

**Human-robot Collaboration in Robotic-assisted
Surgical Training**

YANG TAO

**(M. ENG, NUS)
(B.TECH HONS, NUS)**

A THESIS SUBMITTED

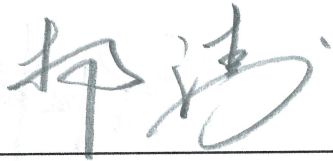
**FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
DEPARTMENT OF MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE**

2015

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

A handwritten signature in black ink, appearing to read 'Yang Tao', is positioned above a horizontal line.

Yang Tao

February 2015

ACKNOWLEDGMENTS

The last four years has been a fruitful and unforgettable journey. Many people came into my life and encouraged me on this work.

Firstly, I would like to express my sincere gratitude to my mentor, Associate Professor Chui Chee Kong, for his continuous inspiration, encouragement and guidance through the course. Prof. Chui is a distinguished researcher with great enthusiasm and passion. He was always there to provide advice with his patience in the last four years. He provided not only advices on research subjects, but also a warm hand to life.

I would like to thank my co-mentor Dr. Liu Jiang for his guidance and open mind over the research work. He has been continuously encouraging me to explore in the research topics throughout the course. I will always remember what Dr. Liu told me in the beginning of this journey: “This is yours, never give it up, no matter what happen in the future”. It encouraged me to overcome many disturbances through the journey.

I would also like to acknowledge the lab head Dr. Huang Weimin and my colleagues from Institute for Infocomm Research for their assistance and support, and creating the lab a very pleasant working environment.

Thanks to my wife and my two sons, my life would not be so colorful without you. Thanks to my parents for their love and sacrifices. Although they had never been voiced, my heart saw them.

TABLE OF CONTENTS

	Page
SUMMARY.....	VIII
LIST OF TABLES.....	X
LIST OF FIGURES.....	XI
LIST OF ABBREVIATIONS.....	XVII
LIST OF SYMBOLS.....	XVIII
1 Introduction.....	1
1.1 Surgical Training.....	1
1.2 Overview of IRAS Training Method.....	4
1.3 Objective and Scope.....	5
1.4 Thesis Contributions.....	7
1.5Thesis Organization.....	7
2 Literature Review.....	9
2.1Medical Simulation.....	9
2.2 Robotics in Surgery and Training.....	12
2.2.1 Robotic-Assisted Surgery and Training.....	12
2.2.2 Surgical Training.....	14
2.2.3 Haptics for Surgical Robots and Simulators.....	17
2.3 Robot Learning from Demonstration.....;	20
2.3.1 Statistical Approach.....	21
2.3.1.2 Hidden Markov Model Approach.....	26
2.3.1.3 Gaussian Mixture Approach.....	28
2.3.2 Neural Networks Methods.....	31
2.4 User Intention Recognition for Human Robot Collaboration	32
2.4.1 Hidden Markov Model.....	34
2.4.2 Probabilistic State Machine.....	37

2.4.3 Dynamic Bayesian Networks Approach.....	42
2.5 Performance Evaluation Methods.....	44
2.5.1 Features for Evaluation Methods.....	47
2.5.2 Evaluation Methods.....	48
2.3.1.1 Hidden Markov Model.....	23
2.5.2.1 Hidden Markov Model for Evaluation.....	49
2.5.2.2 Liner Discriminant Analysis Method.....	51
2.6 Summary.....	51
3 Image-Guided and Robot-assisted Surgical Training System.....	54
3.1 IRAS System.....	54
3.2 Robotic Surgical Trainer for Laparoscopic Surgery.....	57
3.2.1 Design Considerations.....	57
3.2.2 Kinematic Analysis.....	61
3.2.4 Control Hardware.....	64
3.2.5 Control Methods.....	65
3.3Friction Mitigation for Haptic Rendering.....	68
3.4 Experiments.....	71
3.4.1Robotic Performance Analysis.....	71
3.4.2 Experiment of Friction Mitigation for Haptic Rendering	73
3.5 Summary.....	79
4 Motion Modelling, Learning and Guidance.....	80
4.1 Methods.....	81
4.1.1 Data Processing.....	82
4.1.2 Adaptive Mean Shift Clustering of Motion Trajectory	83
4.1.3 Statistical Modelling and Parameter Estimation.....	85
4.1.3.1 Gaussian Mixture Model.....	85
4.1.3.2 Gaussian Mixture Regression.....	87

4.2 Experiments and Results.....	88
4.2.1 Experiments.....	89
4.2.2 Results.....	90
4.3 Discussion.....	95
4.4 Summary.....	101
5 Motion Intention recognition and Its Application in Surgical Training.....	103
5.1 Stacked Hidden Markov Models.....	104
5.1.1 HMM for Motion Intention Recognition.....	104
5.1.2 Stacked Hidden Markov Models.....	105
5.2 Stacked HMM for Laparoscopic Surgical Training.....	106
5.2.1 Observation Features for the HMMs.....	108
5.2.2 HMM Configuration.....	110
5.2.3 HMM Training and Recognition.....	110
5.3 Experiments.....	111
5.3.1 Surgical Simulation and Experiment Design	111
5.3.2 Experiment Evaluation and Discussion.....	115
5.3.2.1 Primitive Layer.....	115
5.3.2.2 Subtask Layer.....	121
5.4 Summary.....	128
6 Surgical Skills Evaluation and Analysis.....	130
6.1 Technical Evaluation.....	131
6.1.1 Evaluation Method.....	132
6.1.2 Experiments.....	135
6.1.3 Performance Analysis and Discussions.....	137
6.2 Clinical Evaluation.....	140
6.2.1 Experiment.....	140

6.2.2 Performance Analysis and Discussions.....	142
6.3 Summary.....	144
7 Discussion And Conclusion.....	146
References.....	149
Author's Publication.....	156

SUMMARY

The paradigm of surgical training has gone through significant changes due to the advancement of technologies. Virtual reality-based surgical training with relatively low cost over long term is now a reality. However, the training quality of such training technologies still heavily relies on the guidance / feedback given by the instructor, normally an expert surgeon, who teaches the user the right surgical techniques. Training quality is subjective to the qualification / experiences of the expert surgeon and his / her availability.

An image guided robot-assisted training system is proposed in this thesis. Our new approach uses a robotic system to learn a surgical skill from an expert human operator, and then transfer the surgical skill to another human operator. This training method is capable of providing surgical training with consistent quality and is not dependent on the availability of the expert. The proposed surgical training system consists of image processing software to construct a virtual patient as a subject for operation, a simulation system to render a virtual surgery, and a robot to learn and transfer the surgical skills from and to a human operator. This thesis focuses on the mechanism of robotic learning and the transferring of surgical skills to human operator and the related topics.

The robotic surgical trainer was designed and fabricated to resemble the tools and operating scenario of a laparoscopic surgery. Tactile sensation is one of the features that a surgeon relies upon for decision making during surgery. Haptic function was incorporated into the robotic surgical trainer to provide user with tactile sensation. The friction of the system is mitigated by motion-based cancellation method for haptic rendering.

In order to enable the robot to learn a surgical skill and provide guidance-based on the learnt skills, the surgical skills need to be generalized and modeled mathematically. A mean shift based method was proposed to identify the motion primitives in a surgical task. Gaussian Mixture Model was then applied to model the surgical skills based on the identified motion primitives and Gaussian Mixture Regression was applied to reconstruct a generic model of the specific surgical skill. Hidden Markov Model method was applied to recognize the intention of a user when he / she was operating on the virtual patient. Proper guidance can be executed based on the recognized motion intention and the general model of the corresponding surgical task.

The proposed surgical training method was evaluated using two experiments. In the first experiment, the performances of two groups of lay subjects are compared. In order to eliminate the subjective bias during the evaluation process, Hidden Markov Model method was applied in the performance evaluation. The second experiment is a clinical evaluation involving medical residents operating on a porcine model. Two groups of residents were trained by the proposed method and conventional method separately, and then operate on the animal. These operations were recorded in video and evaluated by two experienced surgeons. Both studies show that the subjects who underwent the proposed training method performed better than that of the subjects underwent conventional training.

LIST OF TABLES

	Page
Table 2-1 Five-item global rating scale described by Vassiliou et al. [104].	46
Table 2-2 Task-specific checklist presented in [104]: dissection of the gallbladder from the liver bed.	46
Table 3-1 Resolution of the actuators for each DOF	64
Table 3-2 Maximum positional error of each joint.	72
Table 3-3 . Frictional force fitting results with Equation (3.17).	76
Table 4-1 The RMS error of the rotational joints of the reconstructed trajectory to the demonstrations after DTW.	99
Table 4-2 Effect of PCA on RMS error of rotational joints of the reconstructed trajectory to the demonstrations after DTW.	100
Table 5-1 Recognition rates of the primitive recognition model in the frequency domain. Training and test data were represented by different number of Gaussian components. Three HMMs were applied to represent the motion intentions. Each HMM was set with 3 states. Twenty frames were taken for each observation.	117
Table 5-2 Recognition rates of the primitive recognition model in the spatial domain. Three HMMs were applied to construct the recognition model. Each HMM was set with 3 states. Twenty frames were taken for each observation.	117
Table 5-3 Recognition rates of the primitive recognition model in the frequency domain. HMMs in the primitive recognition were configured with different number of states. Intentions were represented by 8 HMMs. Twenty frames were taken for each observation.	118
Table 5-4 Recognition rates of the primitive recognition model in the spatial domain. HMMs in the primitive recognition were configured with different number of states. Intentions were represented by 8 HMMs. Twenty frames were taken for each observation.	118
Table 6-1 Percentages of the observation sequences from Group A and ranked at top N of the 120 observation sequences (N=20, 40, 60).	138
Table 6-2 Participants' performance evaluated by average task time, trajectory length of the left and right instruments.	139
Table 6-3 Surgeon's performance evaluated by average task time, trajectory length of the left and right instruments.	140
Table 6-4 Summary of experiment procedure for clinical evaluation.	142
Table 6-5 Average score of the students in the surgeries.	143
Table 6-6 Average score of 10 subtasks.	143

LIST OF FIGURES

	Page
Figure 1-1 The image guided robotic assisted surgical training method.	4
Figure 1-2 Structure of robot-assisted surgical training system.	7
Figure 2-1 da Vinci surgery system with user handle, and endowrist instrument presented in its website [29, 34].	13
Figure 2-2 Flexible robotic endoscopy [30].	14
Figure 2-3 Endobot for assistance in minimally invasive surgery [31].	14
Figure 2-4 Haptic recording and playback of MISST [45]: (a) A human operator manipulates a laparoscopic tool equipped with sensors, making indentations for measuring and recording interaction forces; (b) a haptic device interfaced with a probe and sensors can be programmed to make controlled indentations for measuring and recording interaction forces; and (c) haptic playback involves the display of programmed forces to a user for guidance and control during training.	17
Figure 2-5 Endoscopic view of maze game presented in [22].	17
Figure 2-6 Bearings on Xitact product for the purpose of supporting and minimizing the friction.	19
Figure 2-7 Information flow for robotic learning by demonstration described by Calinon et al. [61].	22
Figure 2-8 State transition patterns (a) full transition, (b) left to right transition.	
Figure 2-9 Enforced Sub Populations (ESP) neuroevolution method. The Long Short-Term Memory (LSTM) network architecture (shown with four memory cells), and the pseudo inverse method to compute the output weights. When a network is evaluated, it is first presented with the training set to produce a sequence on network activation vectors that are used to compute the output weights. Then the training set is presented again, but now the activation also passes through the new connections to produce outputs. The error between the outputs and the targets is used by ESP as a fitness measure to be minimized (source: [54]).	32
Figure 2-10 Structure of layered hidden Markov Model proposed in [80].	36
Figure 2-11 Two-level layered hidden Markov model implemented by D. Aaron et al. [80], modelling gestemes at level 2 and task a level 1. Level 2 acquires signal from trajectory [80].	36
Figure 2-12 General representation of probabilistic state machines in transition [83].	37
Figure 2-13 Probabilistic state machines for piling and unpling intentions in [83]. (a) State machine for explicitly communicated human intentions (picking and placing an object). (b) Implicitly	38

communicated human intentions.

Figure 2-14	Intention recognition algorithm presented in [83].	39
Figure 2-15	A probabilistic state machine that models the transitions between subtasks in [101].	41
Figure 2-16	An illustration of general form of Bayesian Networks in [102].	43
Figure 2-17	Work path of intention recognition in [103].	
Figure 2-18	Human's intention-action-state flow and DBN corresponding to human intention recognition model with time-delay presented in [84].	44
Figure 3-1	Overview of the IRAS surgical training system: (a) robotic surgical trainer and (b) virtual surgical simulation platform.	55
Figure 3-2	Motion of surgical instrument in laparoscopy procedure.	58
Figure 3-3	(a) Mechanical mobility of the robot. The travelling limit for pitch, yaw, roll, translation, handle grasping motion are 120°, 120°, 360°, 350mm and 60°, respectively. (b) Kinematic model of surgical instrument.	58
Figure 3-4	Details of the robotic surgical trainer.	60
Figure 3-5	Dynamic model of the robot in inertia frame II . l is the distance from the gravity center of tool to the trocar. O_1 is the rotational link attached to the center of the semi-spherical mechanism (see Figure 3-3). O_2 is the translational link attached to the gravity center of the prismatic component.	62
Figure 3-6	Implementation block diagram of the position control for active mode of the robot.	66
Figure 3-7	Block diagram of the force control for passive mode of the robot.	68
Figure 3-8	Control diagram for friction compensation and haptic output. \mathbf{f}_{des} is the desired haptic output reference force, \mathbf{f}_{hap} is the haptic output force, \mathbf{f}_u is the user's interaction force.	71
Figure 3-9	(a) Execution and force on handle of left manipulator, (b) execution and force on handle of right manipulator. Red line is the recorded trajectory, black line is the execution results, blue arrow indicates the force vector on handle, and green arrow indicates the moment vector on handles.	72
Figure 3-10	Relative velocity at the contacting area of each axis. The velocity measured was the relative linear velocity at the contacting area.	73
Figure 3-11	Mean frictional force from current design with haptic output from 1N to 7N. The frictional force is larger when the components are just to move, and it is reduced significantly and tends to stabilize when the components moving at higher velocity. The frictional forces are generally higher when the robot outputs a higher haptic force. Vertical bars are the standard deviations at the specific velocity and haptic output. (a) Frictional force for pitch axis. (b) Frictional force for yaw	75

axis.

Figure 3-12	Surface fitting result with Equation (3.17). Experimental results shown in Figure 3-11 were fitted with Equation (3.17) using Matlab curve fitting toolbox. Black dots are the down sampled experimental measurements. The meshed surface is the fitting results. (a) Frictional force fitting for pitch axis with $\mu_e = 0.08$, $a_2 = -0.032$, $a_1 = 0.403$, $a_0 = 1.476$. (b) Frictional force fitting for yaw axis with $\mu_e = 0.086$, $a_2 = -0.019$, $a_1 = 0.351$, $a_0 = 1.82$.	76
Figure 3-13	Mean residual frictional force measured with compensation. Vertical axis is the measured frictional force after compensation. Vertical bars are the standard deviations at the specific velocity and haptic output. (a) Frictional force for pitch axis. (b) Frictional force for yaw axis.	78
Figure 4-1	Data processing procedure to model and reconstruct the motion trajectories.	81
Figure 4-2	A tissue division simulation on a virtual patient in the surgical simulation system.	90
Figure 4-3	Comparison of the raw motion data collected from the simulator and the motion data after multi-dimensional Dynamic Time Warping. (a) and (c) are raw motion data, (b) and (d) are the motion data after DTW. The circled sections indicated the overlapped features in the raw motion data, and the results after DTW.	90
Figure 4-4	The adaptive bandwidth value for the left and right trajectories of the instrument in one demonstration.	91
Figure 4-5	The GMM modelling and the GMR regression results based on the proposed method. (a) and (c) are the GMM encoding for the tissue division task of the left and right instruments, respectively, based on the adaptive mean shift clustering results. The spot is the mean of each Gaussian component, and the patch is the square root of covariance matrix of the corresponding Gaussian component. (b) and (d) are the GMR regression results, the solid line is the expected mean of each Gaussian model at the given time t , and the patch is the expected square root of the covariance matrix at the given time t .	93
Figure 4-6	Raw motion trajectories and mean reconstructed model of Subject 1: (a) 22 motion trajectories (positional only) of the surgical tool tip in the tissue division task, (b) reconstructed mean trajectory by GMM and GMR. The orientation of instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument. The arrows indicate the direction of motion.	94
Figure 4-7	Raw motion trajectories and mean reconstructed model of Subject 2. (a) 22 motion trajectories (positional only) of the surgical tool tip in the clip deployment task. (b) Reconstructed mean trajectory by GMM and GMR. The orientation of	94

instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument.

Figure 4-8	Raw motion trajectories and mean reconstructed model of Subject 3: (a) 24 motion trajectories (positional only) of the surgical tool tip in the clip deployment task, (b) reconstructed mean trajectory by GMM and GMR. The orientation of instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument.	95
Figure 4-9	GMM modelling based on K-means method and fixed bandwidth mean shift method. (a) and (b) are the GMM modelling results with K-means clustering method for the left and right instruments trajectories, respectively. (c) and (d) are the GMM modelling results with fixed bandwidth clustering method for the left and right instruments trajectories, respectively.	98
Figure 4-10	The GMM modelling results of the tissue division trajectory without the PCA analysis. The data across large time span were grouped in same motion primitive. (a) and (b) are the GMM modelling of trajectories for left and right instruments, respectively.	100
Figure 5-1	Stacked HMMs for motion intention recognition.	106
Figure 5-2	Stacked HMMs for motion intention recognition in laparoscopic surgical training.	107
Figure 5-3	Observation sequence formed by data windows.	111
Figure 5-4	(a) Grasping forceps and hook, (b) Curved forceps, (c) Clip applicator, (d) Scissors.	113
Figure 5-5	(a): State diagram for the surgical procedure; (b) (c) (d) (e): state diagrams for the respective sub tasks in Section 5.3.1.	114
Figure 5-6	Effects of HMM numbers to recognition rate in primitive recognition model. HMMs were configured with 3 states, with 3 Gaussian components for observation sequence. (a) Recognition rates in the frequency domain. (b) Recognition rates in the spatial domain.	116
Figure 5-7	Sample of recognized intention for left (a) and right (b) motion trajectory in the frequency domain. Eight HMMs were used to represent the intention. Each HMM was set with 3 Gaussian components and 3 states.	119
Figure 5-8	Sample of recognized intention for left (a) and right (b) motion trajectories in the spatial domain. Eight HMMs were used to represent the intention. Each HMM was configured with 3 states and data were modeled by 3 Gaussian components.	120
Figure 5-9	Recognition rate of the subtask recognition model trained based on primitive recognition model in the spatial domain. Primitive recognition models were constructed by 8 HMMs.	123

	Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.	
Figure 5-10	Recognition rate of the subtask recognition model trained based on primitive recognition model in the spatial domain. Primitive recognition models were constructed by 3 HMMs, Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.	123
Figure 5-11	Recognition rate with respect to the width of observation window. Red lines: the subtask recognition model was configured with 7 states, three Gaussian components and the respective primitive recognition model was constructed with 8 HMMs. Blue lines: the subtask recognition model was configured with 13 states, nine Gaussian components, and the respective primitive recognition model was constructed by 3 HMMs.	124
Figure 5-12	Sample of recognized motion intention in subtask level. The subtask recognition model was configured with 7 states and 3 Gaussian components. (a) Normalized log likelihood of four subtasks, (b) recognition result.	125
Figure 5-13	Recognition rate of the subtask recognition model trained based on primitive recognition model in the frequency domain. Primitive recognition models were constructed by 3 HMMs. Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.	126
Figure 5-14	Recognition rate of the subtask recognition model trained based on primitive recognition model in the frequency domain. Primitive recognition models were constructed by 8 HMMs. Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.	127
Figure 5-15	Recognition rate with respect to the width of observation window. Red lines: the subtask recognition model was configured with 9 states and 9 Gaussian components and the respective primitive recognition model was constructed with 8 HMMs. Blue lines: the subtask recognition model was configured with 7 states and 7 Gaussian components, and the respective primitive recognition model was constructed by 3 HMMs.	127
Figure 6-1	Instruments' tip to the specified points on the organ, PO_L and PO_R ; relative position vector from the left instrument's tip to the right instrument's, P_{LR} ; angle between the instrument's tip vector and the specific vectors on the organ, β_L and β_R ; angle of the instrument handle opened, α_L and α_R , they are proportional to the angle of applicator's jaws formed.	133
Figure 6-2	Three states full transition HMM. π is the prior probability, a is the state transition matrix and b is the observation probability.	134
Figure 6-3	Convergence of log likelihood for each subtask in parameter estimation process.	135
Figure 6-4	Mean log likelihood and standard deviation of each test	138

session. Dark solid lines represent Group A, dash lines represent Group B and vertical bars represent the standard deviation of the likelihood for each test session.

Figure 6-5 Laparoscopic training boxes for Control Group's training.

141

LIST OF ABBREVIATIONS

BIC	Bayesian Information Criterion
BN	Bayesian Network
CDF	Cumulative Distribution Function
CPD	Conditional Probability Distribution
CT	Computed Tomography
DBN	Dynamic Bayesian Network
DTW	Dynamic Time Warping
ESP	Enforce Subpopulation
GMM	Gaussian Mixture Model
GMR	Gaussian Mixture Regression
HGF	Haptic Guidance by force
HGP	Haptic Guidance by position
HMM	Hidden Markov Model
IRAS	Image guided robotic assisted Surgery
LDA	Linear Discrimination Analysis
LSTM	Long Short-Term Memory
MIS	Minimally Invasive Surgery
PCA	Principal Components Analysis
PSD	Power Spectrum Density
RNN	Recurrent Neural Network
RMS	Root Mean Square
SVM	Support Vector Machine
VR	Virtual Reality

LIST OF SYMBOLS

a^*	Expected action
a_2, a_1, a_0	Coefficients of polynomial function
a_{ij}	Transition probability of i to j
\mathbf{A}	Transition matrix
\mathbf{A}_p	Transformation matrix
b_{ij}	Observation probability
\mathbf{B}	Observation matrix
$B_i(w_i \theta_i)$	Belief function
c_1	$\cos(q_1)$
c_{rr}	Rolling resistance coefficient
d	Dimensionality
\mathbf{D}	Mass inertia matrix
e_f	Force error
E_{k_p}	Cumulated posterior probability
\mathbf{f}_c	Coulomb frictional force
\mathbf{f}_d	Haptic cue force
\mathbf{f}_{des}	Desired haptic output force
\mathbf{f}_{dis}	Non-rigid force
\mathbf{f}_e	Force acting with the environment
\mathbf{f}_{hap}	Haptic output force
\mathbf{f}_N	Normal force
\mathbf{f}_{roll}	Rolling frictional force
\mathbf{f}_s	Stribeck force
\mathbf{f}_{ss}	Steady state friction
$f(\mathbf{f})$	Joint multivariable distribution function
f_x, f_y, f_z, f_g	Forces in x, y, z and g directions
F_A	Final state probability
F_R	Status of footswitch
\mathbf{g}	Potential energy vector

h	Bandwidth
h_o	Initial bandwidth
H_k	Hamming window with width k
\mathbf{I}	Inertia tensor
I	Second Moment of inertia
I_A	Initial state probability
II	Inertia frame
\mathbf{J}	Jacobian matrix
k_{hap}	Environmental stiffness for haptics
K_p	Number of mixture components
L	Likelihood of the model
L_m	Mean log-likelihood
\mathbf{m}_v	Mean shift vector
M	Number of trials
n_p	Number of free parameter
N	Number of observations
O	Observation sequence
O_1	Rotational link
O_2	Translational link
$O_{Left.k}$	Observation sequence formed by the k^{th} cluster in the data collected from left hand instrument
$O_{Right.k}$	Observation sequence formed by the k^{th} cluster in the data collected from right hand instrument
O^{S2}	Observation sequence from expert surgeon 2
\hat{O}^S	Optimal observation sequence generated by the expert surgeon's model λ_S
O_{st}	Observation sequence for subtasks in evaluation.
O_{sub}	Observation sequence for subtask recognition model.
P	Position of the tip of instrument
P_A	Transition probability
P_L	Normalized observation probability for left hand
P_R	Normalized observation probability for right hand

q_1, q_2, q_3, q_4	Joint variables of the robot
q_d	Desired trajectory
s_1	$\sin(q_1)$
$s_{i,t}$	Subtask state
S	Optional state transition sequence
S_m	Evaluation metric
\mathbf{t}_n	Current position of the manipulator
\mathbf{t}_m	Predetermined trajectory
\mathbf{T}	Motion trajectory in latent space
\mathbf{u}	Coriolis and centrifugal force vector
v	Relative velocity of two moving components
v_s	Stribeck velocity
V	Velocity of the tip of surgical instrument
w	Weight vector
\mathbf{x}_i	Trajectory data
$\hat{\mathbf{x}}_t$	Conditional expected time
\mathbf{x}	Conditional expected motion trajectory
$\hat{\mathbf{x}}_s$	Conditional expected spatial vector in motion trajectory
\mathbf{X}	Motion trajectory
z_k	State k of Hidden Markov Model
\mathbf{Z}	Fourier Transform of observation sequence
α	Acceleration of tip of instrument
δ_A	Transition set
ϕ	Force coefficient
Γ	Mixture coefficient
λ	Hidden Markov Model
$\bar{\lambda}$	Estimated Hidden Markov Model
λ_L	Recognition model for left hand instrument
λ_R	Recognition model for right hand instrument
λ_{st}	Hidden Markov Model trained by expert surgeon's data

λ_{sub}	Subtask recognition model
μ	Mean vector
μ_e	Equivalent frictional coefficient
μ_s	Sliding frictional coefficient
τ_x, τ_y, τ_z	Moment about x, y and z direction
v	Eigenvector
π	Prior probability
θ_t	Game history
Ω	A set of actions in Probabilistic State Machine
$\xi^x(t)$	Path of hand motion in latent space
$\xi^\theta(t)$	Joint angle in latent space
ψ	Proportionality constant
Σ	Covariance matrix
Σ_{k_p}	Covariance matrix of k_p^{th} component
Σ_{ss}	Conditional expected covariance matrix of spatial vector

1 INTRODUCTION

Surgical training is one of the key components in the life of a medical staff. "See one, do one, teach one" [1] used to be a common technique in surgical training. However, training strategies have been changed in the past decades due to the advancement of surgical techniques, robotic and computer simulation technologies. The medical education providers are expected to enable the students to "see one, simulate many, do one competently, and teach everyone" [1].

Robotic technologies have been widely applied in surgery. It has been playing a significant role in robot-assisted surgery, teleoperation [2, 3] and robotic surgical training [4, 5]. Many of the technical limitations of surgery might be circumvented with the advent of robotic technologies [3]. Researchers have explored the application of robotic assistance to train motor skills, such as teaching calligraphy [6, 7]. However, robotic assistance for the honing of surgical skills, especially laparoscopic motor skill, to our knowledge, has not been well studied. In this thesis, the robotic technologies and its applications in laparoscopic surgical training involving human-robot collaboration are explored.

1.1 Surgical Training

The challenges of open surgery and Minimally Invasive Surgery (MIS) are different. A surgeon who performs MIS is required to confront challenges in open surgery and challenges associated with MIS, such as hand-eye coordination and depth perception.

Laparoscopy surgery is a minimally invasive surgical techniques commonly used for many abdominal surgeries, including cholecystectomy (removal of gallbladder for stone and other disease), liver tumour treatment (ablation, resection etc.), pancreas surgery, gastrointestinal (stomach and large intestine) and urologic surgery. Laparoscopic surgery provides several major benefits to the patients as compared to open surgery, such as shorter recovering time and smaller scar. It has been widely adapted in clinical practice due to the benefits that this technology brought to the patients. Ninety-five percent of cholecystectomy was performed laparoscopically as reported in [8]. However, Laparoscopic surgery could benefit the patients only on the condition that the surgeons are competent to perform the laparoscopic surgery safely. Therefore, laparoscopic surgery is suggested to be performed by experienced surgeons [9].

There are many natural constraints inherent to laparoscopic surgery. Intensive training is required to overcome the natural constraints imposed on the surgeon, and it is crucial for the surgeon to obtain the necessary level of proficiency to perform laparoscopic surgeries safely and effectively [10]. Traditionally, such surgical training is done with the 'master-apprentice' strategy. However, the traditional strategy and the "See one, do one, teach one" method could not meet the requirement in acquiring laparoscopic skills considering patient safety.

With advanced computer simulation and virtual reality technologies, various surgical simulators, such as LapVR [11], Lap Mentor [12] and RoSS [13], are available for the surgical students to practice with a generic anatomic model. These simulators provide a good practicing environment for novice surgeons to practice their skills. Although some supervising features are

available in these simulators, such as audio, text or video, the novice surgeons or residents have to fine tune their skills through practicing on real patients under the guidance of the experts in the operating theatre. With increasing complexity of the surgical operations, it becomes increasingly dangerous for the novice surgeons to ‘learn’ and gain experiences while operating on a real patient despite being supervised during the operation. The experienced surgeon may teach the novice surgeons by holding and guiding their hands to perform tasks or corrections in order to train their motor skills in the operating theatre. Although there are various advantages associated with each of the simulators / training methods, none of them mimic the conventional ‘hand-by-hand’ guidance that surgeons applied in the operating theatre.

Physical guidance plays an important role in the surgical training where the experienced surgeon corrects the motion of the novice while conducting a procedure. Physical guidance is necessary when the novice surgeon learns how to use the surgical tool and the necessary techniques to conduct a specific procedure. ‘Hand-by-hand’ guidance training strategies are reliable and effective techniques in laparoscopic surgical training, especially for difficult surgical scenarios. However, this type of training strategy is time consuming for the experienced surgeons to teach every medical student ‘hand-by-hand’ in the training course. This training strategy also introduces risks to the patient in the operating theatre. Unfortunately, there are no other means for a novice surgeon to gain expertise and become an expert besides gaining experiences by practicing on real patients. Therefore, a training media is required to bridge the gap by acquiring the expertise of the experienced surgeons and physically guide the novice surgeon for training. A new surgical training method was

proposed and developed to bridge such a gap - the image guided robotic assisted surgical (IRAS) training method which is capable of acquiring surgical skills and guiding the novice in honing their motor skills for laparoscopic surgery.

1.2 Overview of IRAS Training Method

The platform of the IRAS training method includes a patient-specific virtual patient and a robot-assisted surgical training system, as shown in Figure 1-1. The patient-specific virtual patient provides a training object for the robotic surgical trainer and trainee to operate on. The robotic surgical trainer plays two roles in the system: (1) it is a learning platform in between the experienced surgeon and virtual patient; and (2) it is also a teaching interface in between the trainee and the virtual patient that he / she is practicing upon.

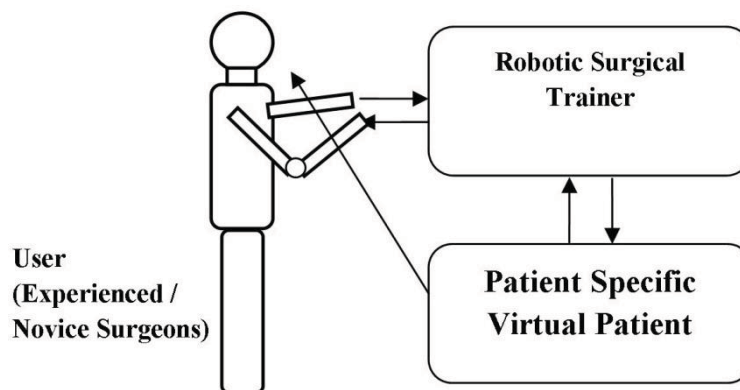


Figure 1-1 The image guided robotic assisted surgical training method.

The IRAS training method is designed to facilitate surgical training with the robotic learning methods to achieve similar outcome as that of ‘hand-by-hand’ physically guided training. There are two modes in the IRAS train system: *Acquisition Mode* and *Guidance Mode*. In the *Acquisition Mode*, the master operates on a 3D virtual patient model which is reconstructed from patient’s

Computed Tomography (CT) images, and has his / her hand motions recorded and learned by the IRAS training system. Complete guidance and haptic cue guidance are provided by the *Guidance Mode*. In the complete guidance mode, the robotic surgical trainer replays the acquired instrument manoeuvre (such as trajectory of the instrument), and the novice surgeon experiences the tool manipulative motion of a master surgeon kinaesthetically by holding onto the surgical instrument. This provides a deeper appreciation to the master surgeon's motion than mere visual and didactic guidance. In haptic cue guidance, the novice is allowed to operate on the patient-specific anatomical model based on his / her own knowledge. The robotic surgical trainer provides him / her with some degrees of motion guidance, i.e. haptic could imply through the robotic surgical trainer if the novice's operation deviates severely from the experienced surgeon's operation. It is advantageous that the novice surgeon can be trained via 'hand-by-hand' method without the experienced surgeon being physically presented in the training premises. Although there has not been any conclusive evidence of benefits to laparoscopic training through kinaesthetic guidance from recorded motion, subjects appear to perform tasks better after going through it as suggested in [14].

1.3 Objective and Scope

Robot-assisted surgical training is a machine-mediated motor skill training system which uses the similar concept of robotic-assisted teaching calligraphy [6, 7]. Robotics assistance in laparoscopy surgery training can transfer the skills of experienced surgeons to the novice surgeons as the experienced surgeons do, and reduce the working load of experienced surgeons in training.

The objective is to research, develop and experiment robot-assisted surgical training through the IRAS system. In order to achieve the goal of the IRAS training method, a new robotic mechanism was designed, developed and examined. The robot is required to have the capability of acquiring knowledge on manoeuvring of surgical instrument from the human. With the knowledge acquired by the robot, the robot should have the capability to recognize the intention of user / novice surgeon when the robot observes a novice surgeon performing the task which the robot has the knowledge of. Robotic learning and intention recognition are two challenging tasks in developing the IRAS training system. A robotic mechanism has been designed and developed for the IRAS, and it has been investigated in the three main components: mechanism, machine learning and human-robot collaboration.

Figure 1-2 shows the overall structure of the robotic surgical trainer. It consists of the robot hardware, the knowledge representation and the human robot collaboration. The hardware has been designed for a specific category of task with both input and output mechanisms to interact with the environment / user. In the knowledge representation part, A framework that enable the robot to acquire knowledge of the skills and represent the skill with mathematical models was developed. In the human robot collaboration part, an intention recognition method was developed that enables the robot to realize the intention of the user based on the motion trajectory while the user performs a surgical task.

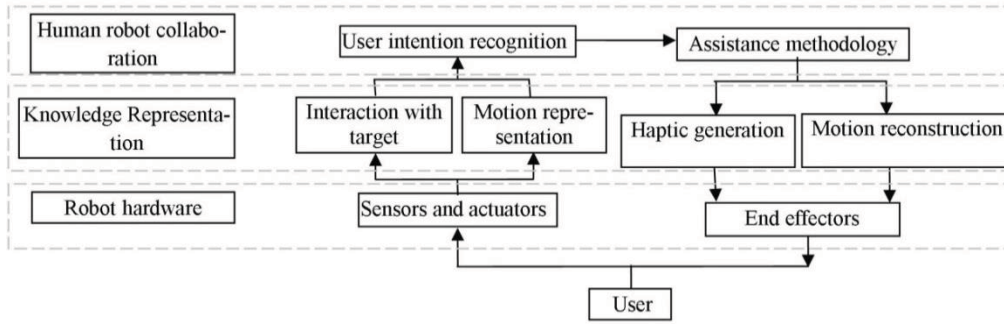


Figure 1-2 Structure of robot-assisted surgical training system.

1.4 Thesis Contributions

The contributions of this thesis are as follows:

- An innovative robotic surgical trainer was proposed and developed. This robotic surgical trainer has been awarded a US patent 8,764,448 on 1st July 2014;
- A new framework for robotic learning based on demonstration was proposed to model a surgical skill;
- A stacked Hidden Markov Model method was proposed to recognize motion intention in surgical training; and
- Experimental results demonstrated the feasibility of the proposed surgical learning method.

1.5 Thesis Organization

This thesis discusses the important technical components and engineering challenges in the IRAS training system. The thesis is organized as follows:

- Chapter 2 reviews the components in the IRAS training system, including medical simulation, robotics in surgery and training, robot learning from demonstrations, motion intention recognition, and surgical performance evaluation method;

- Chapter 3 describes the system design and mechanism for surgical training in details;
- Chapter 4 presents a new framework in identifying the motion primitives from surgical motion trajectory;
- Chapter 5 discusses the issues raised to recognize the intention of the users when the user operates on the virtual patient during training;
- Chapter 6 presents the effectiveness of the IRAS training system evaluated through technical study and clinical evaluation; Finally,
- Robotic-assisted surgical training method was discussed and concluded in Chapter 7.

2 LITERATURE REVIEW

The three main components for building a robotic surgical trainer are robot mechanism, machine learning and human-robot collaboration. This chapter reviews the existing works for the above three components associated with medical simulation, robotics and surgery, and surgical skill evaluation.

2.1 Medical Simulation

Medical simulation is a branch of simulation technology related to education and training in medical fields. It includes simulated human patient [15, 16], simulated clinical environments [15] and simulated task trainers [16]. The main purpose of medical simulation is to train medical professionals to reduce accidents during surgery, prescription, and general practice. Simulation provides medical educator with a controlled training environment under a variety of circumstances, such as uncommon or high-risk scenarios. Nowadays, patients increasingly concern on medical students practicing on them. Medical educator has faced the challenges by restructuring curricula, to bridge the gap between the classroom and the clinical environment. Medical simulation has been the solution identified to bridge the gap. Simulation-based training was demonstrated to lead to clinical improvement in two areas [17]. One of the areas is that those residents trained on laparoscopic surgery simulators showed improvement in procedural performance in the operating room [17]. Simulated human patient and simulated task trainer are reviewed in the following section.

Mannequin and computerized virtual patient are two major simulated human patients for medical education. Mannequin used to be a simple training device for medical educators. With the advancement of technologies, the

training devices have changed from simple organ models to high-fidelity mannequin simulators. These mannequin simulators are equipped with life-like features which are capable of recreating physical examination findings, such as normal and abnormal heart and lung sounds, pupil diameter, sweating, and cyanosis, as well as physiological changes, such as changes in blood pressure, heart rate, and breathing [17]. The mannequin simulators may be designed for general physical examination purpose and for specific tasks, such as the Endovascular Simulator [18] which is for endovascular surgery simulation. These high fidelity mannequin simulators assist the user in understanding the anatomy, pathological reaction of patients, and hence, have improved the quality of medical education.

Virtual patient is another important innovation that has advanced medical education. The visible human project laid a great foundation for computerized virtual patient which has been applied as part of simulated task trainer. Simulated task trainers are commonly seen for surgical training. There are a wide range of such simulators in both the research and commercial market [19, 20]. It ranges from camera-based training box to virtual reality based simulators, to robotic-assisted training devices [21-23].

Simulated human patient is used in the simulated task trainer as a medical object. However, for the simplicity and stability of the simulation system, generic human patient model are usually employed. With the advances in image processing, computer graphic and 3D reconstruction technologies, patient-specific simulator are getting more attentions from researchers [19], such as patient-specific simulator for cerebral artery [24], plastic surgery [25], fracture surgery [26], laparoscopic colectomy [27] and carotid artery stenting

[28], etc. The development of such medical simulation involves medical image processing which extracts the anatomical information from the CT / MRI images; 3D reconstruction of anatomical model; simulation of deformation; recording and evaluation of simulated procedure. Each component links to numerous interesting research challenges to be explored.

Patient-specific simulators bring incomparable benefits to the medical student, patient, and the medical staff. The patient-specific simulator allows preoperative rehearsal of actual and upcoming patient cases on the simulator. These simulators bring the virtual reality (VR) simulation concept of simulated rehearsal to allow practice of a specific event. It is a great improvement of VR comparing merely acting as a generic training tool to practice for a specific skill. The patient-specific simulators not only allow procedure planning but also allow a 'hands-on' rehearsal of the actual procedure [19]. Hence, user can conduct both cognitive rehearsal and psychomotor rehearsal. These characteristics could enhance patient's safety by boosting the level of physician's preparation work, and preventing complications or suboptimal surgery.

However, all these training technology only provides a tool to the medical students. To achieve better training quality, it still relies on the coaching from the experienced medical staff, i.e. surgeons. The surgical training process is labour intensive from the perspective of medical staff. The training quality is subjected to the quality of the expert surgeon.

2.2 Robotics in Surgery and Training

2.2.1 Robotic-Assisted Surgery and Training

Robotic-assisted surgery is an application of robotics in medicine with an aim to assist clinicians during surgery. Many surgical robots have been built for Minimally Invasive Surgery (MIS) [29]. The robots have played different roles in robotic-assisted surgery. They can be divided into two groups according to their roles:

1. Master-slave robot system. These robots emulate the dexterous motion of the surgeon's hand movement from a user-comfort space to a confined space. Such as da Vinci [29] and Flexible Robotic Endoscopy [30]; and
2. Assistive robots. These robots focus on providing assistance during surgery, such as constraining or enhancing the mobility of surgical instrument's motion [31, 32], or performing some repetitive tasks like suturing [32, 33].

Both da Vinci and Flexible Robotic Endoscopy [30] have a control console and a robotic mechanism to accept the surgeon's input motion, and an end effector which directly operates on patient's pathological site, as illustrated in Figure 2-1 and Figure 2-2. This type of robotic-assisted surgery extends the Degree-of-Freedom (DOF) of traditional surgical tools, and hence releases the potential of surgeon's technical capability in a confined space. These robots require innovative design on both user end (master robot) and surgical end (slave robot). The robots can also be built with haptic feedback function at the user end to give the user a sensation of palpation. However, they do not

provide direct guidance on how to improve the quality of operation, for example guiding the user to cut according to a planned path or avoiding certain places.

Endobot [31], as shown in Figure2-3, is capable of restricting the motion of the surgical tool, and moving the surgical tool according to a predefined profile or within a predefined zone for controlling the tissue cutting procedure. Liu et al. [32] studied the robot for reducing the tremor of the surgeon's hand in vitreoretinal surgery. The robotic system can minimise the damages on the optic nerve. Hermann et al. [33] describes a robot that can learn surgical knot tying with laparoscopic surgical instrument. The second group of robots focus on robot learning and human robot collaboration.

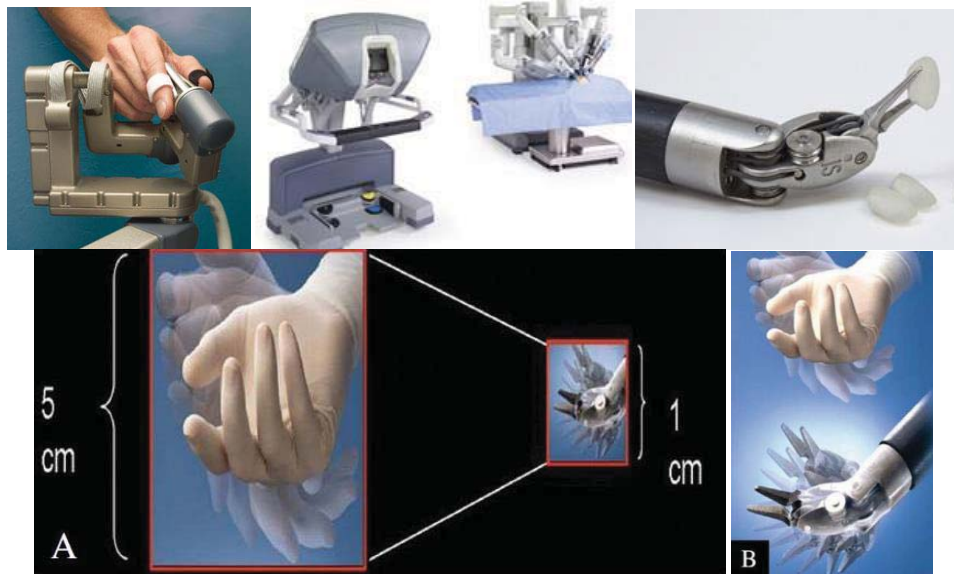


Figure 2-1 da Vinci surgery system with user handle, and endowrist instrument presented in its website [29, 34].



Figure 2-2 Flexible robotic endoscopy [30].



Figure 2-3 Endobot for assistance in minimally invasive surgery [31].

2.2.2 Surgical Training

Surgical training is a significant component in the career of a clinician. The learning curve is long for a novice surgeon to execute the surgical skills at

certain proficiency level. A successful surgeon may also spend a large amount of time in conveying his / her skills to the next generation of surgeons.

Surgical skills consist of theoretical skills and practical skills. Theoretical skills are often taught and tested through classroom and examinations. Practical skills are acquired through motor skill training. Learning motor skills is an iterative process of improving the performance [35]. Mark et al. [35] found that verbal feedback and demonstrations from experienced surgeon is more effective than self-assessed feedback of motion efficiency in learning new surgical skills. In their study, one group of students were given verbal feedback and demonstrations regarding to the training skills. This group of students demonstrated good retention of skill when they were tested one month later [35]. The surgical skills can be effectively taught through demonstration and physical guidance. Currently, the medical students, residents or novice surgeons practice on surgical simulators [20, 36-40] and cadavers to gain and fine tune their motor skills. However, all these simulators provide rather a practice environment rather than servicing as an active teaching tool.

With developed robotic technologies, researchers have devoted a lot of efforts in robotic-assisted methods for motor skill training, such as handwriting training [6, 41-43]. There are two types of robotic-assisted motor skill training described in [7]:

1. Haptic Guidance by position (HGP) which uses position information of trajectory for guidance to learn; and
2. Haptic Guidance by force (HGF) which uses force generated by teacher to control the training's performance (HGF).

Teo et al. [6] applied a robotic guidance method to teach Chinese cartography. They applied both types of robotic-assisted motor skill training, i.e. motion guidance (HGF) and path guidance (HGP). Both guidance methods utilised the writing trajectory acquired from the experts. Some freedom was given to the user to follow the trajectory and hence learn the motor skills. Wang et al. [42] applied both haptic and graphic cue methods in the teaching of handwriting. The characters used for motor skill training is a computer generated model. They proposed that a combination multimedia is more effective for motor skill training [42, 44].

Researchers have also devoted efforts in robotic-assisted surgical training, such as medical simulators and robotic surgical training systems. Basdogan et al. [45] developed a robot surgical training system (MISST) with haptic guidance. They applied haptic feedback to guide the user to move along the pre-recorded trajectory. Figure 2-4 illustrates the haptic recording and playback method in [45].

Lee et al. [22] applied robot-assisted training method to train subjects with fundamental laparoscopic surgical (FLS) skills through maze games, as shown in Figure 2-5. The subjects were divided into two groups. The first group performed FLS training on their own without guidance. The second group received guidance from a pre-recorded expert's performance. The experimental results showed that the second group which received guidance achieved a performance closer to that of the expert's performance in terms of spatial and temporal derivation. However, this test was only conducted for FLS skills training.

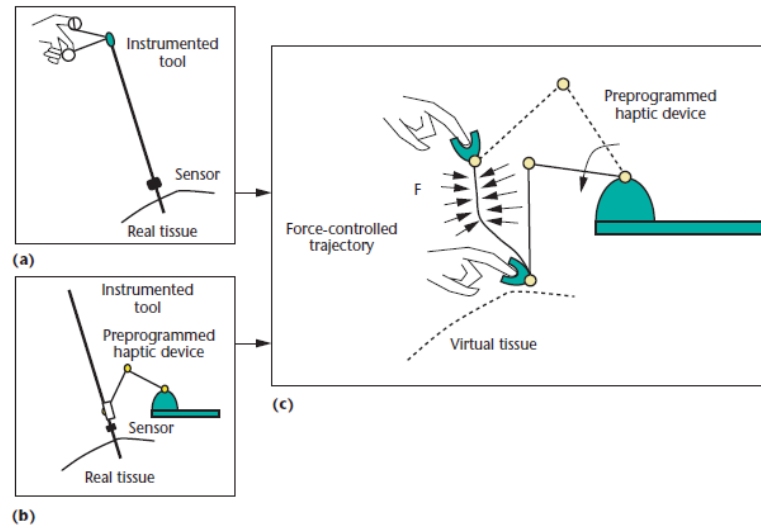


Figure 2-4 Haptic recording and playback of MISST [45]: (a) A human operator manipulates a laparoscopic tool equipped with sensors, making indentations for measuring and recording interaction forces; (b) a haptic device interfaced with a probe and sensors can be programmed to make controlled indentations for measuring and recording interaction forces; and (c) haptic playback involves the display of programmed forces to a user for guidance and control during training.



Figure 2-5 Endoscopic view of maze game presented in [22].

2.2.3 Haptics for Surgical Robots and Simulators

Hardness, color and morphology of the pathology site are the important cues in a laparoscopic surgery. Tactile sensing helps the surgeon to perceive the hardness of the pathology site. The tactile information conveys the tool-tissue interaction status to the surgeon through the sense of touch. This information always plays an important role in decision making during surgery.

In MIS, the surgeon has limited access to the pathological site. The tactile feedback provides information not only on pathology but also the depth of the MIS instruments. During a training process, the training instructor also teaches the medical residents to perceive the tactile information. There are several simulators with haptic feedback function available in the market or have been studied by the researchers, such as Xitact [46], Lap Mentor [47], EndoBot [31] and Sofie force-feedback surgical robot [48]. Different simulator or robotic-assisted surgery systems use different mechanical designs to facilitate the haptic function. However, the famous surgical robot da Vinci [49] was not built with haptic function initially. Nowadays, due to the advent of computers, robotic and virtual reality technologies, various types of simulators and robotics assisted surgery and training devices have been developed for MIS surgical training purpose. Most of the surgical simulators and surgical robots are designed with haptic output capability that enables the system to give tactile feelings to the user.

Xitact [46], developed by Mentice, is a haptic simulation hardware for minimally invasive surgical procedures, such as laparoscopy, nephrectomy, arthroscopy, and even cardiac surgery. Xitact makes the haptic medical simulation realistic and real-time. Action and reaction are synchronized so that the resistance of virtual organ is recreated in the ‘touch’ sensations experienced by the user. The Xitact applied a unique mechanical design in moving the instrument about the trocar point. The haptic output of Xitact is generated by motors actuators and transmitted by strings and linear bearings. Ball bearings and linear bearings were applied at its moving joint to reduce

friction in the design, as shown from a picture of its product in Figure 2-6. Lap Mentor series of products [47] use the Xitact as its actuator in its simulator.



Figure 2-6 Bearings on Xitact product for the purpose of supporting and minimizing the friction.

EndoBot [31, 50], developed by Rensselaer Polytechnic Institute, was designed to assist surgeons during minimally invasive procedures. The robot is able to assist the surgeon to complete some specific tasks, such as suturing, or constraining manual suturing in certain path. Major components of the robot tilting on top of a spherical mechanism, sliding friction occurred in between of the moving components. Sofie [48] is another surgical robot developed by the Eindhoven University of Technology. It made use of a serial robot to assist surgeon in performing tasks. Haptic output is given to the user while performing the tasks. Friction on each robot joint has been minimized and compensated for haptic feedback. MASTER [30, 51] is a master and slave transluminal endoscopic robot developed by Nanyang Technological University of Singapore. Haptic output is given at the master robot so that the user could have a tactile feeling. The master robot was constructed by a serial

robotic mechanism. There was no special consideration on friction control in the design [30].

2.3 Robot Learning from Demonstrations

The learning of human gestures by imitation for a humanoid robot has attracted intensive research efforts in the past decades [52, 53]. Researchers have devoted their efforts into enabling robots to perform motions or manipulations like a human does.

Demonstration-based learning techniques [53] are methods that enable a robot with a motion learning capability. It is achieved through modelling the demonstration; and then reconstructing an optimal motion trajectory for a robot. Research in demonstration-based learning has long been studied in the field of neural networks [54-58] and statistical representation [59-65].

Reinforcement learning methods have also been implemented for motor skills learning for robotics [55-58]. However, the motor skills learned by reinforcement learning do not encode the character of human's motion. It is successful in programming robot's motor skills but these motor skills might not be suitable to be taught to humans, especially for surgical skills. Therefore, the reinforcement learning methods are not discussed here.

Various statistical methods have been explored by researchers, such as the Hidden Markov Models (HMM) [60, 66-69] the Gaussian Mixture Modules (GMM) [59, 61] and the node transition graphs method [62]. Mayer et al. [54] applied a recurrent neural network to learn the tying of surgical knots based on a human operation trajectory.

In recent research, learning by demonstration technology has also been applied in robot-assisted surgery in helping to recognize, learn and evaluate the motion trajectory of a surgical instrument during surgery. Reiley et al [59, 63] applied a statistical modelling method in learning and categorizing motion in surgery. Lin et al. [64] applied Linear Discrimination Analysis (LDA) and Bayes classifier methods in motion modelling for the purpose of skill evaluation in robot-assisted surgery. The following subsections review the various methods discussed above.

2.3.1 Statistical Approach

Hidden Markov Models and Gaussian Mixture Models are the common statistical approaches utilized in Learning from Demonstration. The general data flow for statistical approach of learning from demonstration is described in Figure 2-7. It includes the following steps: reduction of dimensionality, probabilistic data encoding, determination of task constraint, reconstruction of optimal trajectory and recovery of dimensionality.

Principal Components Analysis (PCA) can be applied to identify the principal axis of data and reduce the dimensionality. In order to model the motion acquired from demonstrations, it is essential to identify the motion primitives. Motion primitive is a motion segment which contains the similar characteristics of motion, such as velocity, acceleration and direction. Clustering of demonstrated motion is required before that statistical modelling approach. In a HMM or GMM based approach, a left to right model or single-chain cyclic model with a predefined number of motion primitives can be assumed, and arbitrary numbers of interconnected motion primitives are not considered [62]. Different methods have been proposed to identify the number of mixture

components to model the motion, such as the cross validation, the Akaike information criteria and the Bayesian information criteria. The cross validation method requires independent trials of demonstration to form a complete test set. Calinon et al. [61, 70] applied Bayesian Information Criterion (BIC) scoring method to determine the optimal number of clusters / states. The BIC scoring method is a trade-off of a log-likelihood and the number of parameters to model the motion. The clusters / states are decided by the BIC function Equation (2.1) which gives the lowest score [61]:

$$S_{BIC} = -\log(L) + \frac{n_p}{2} \log(N), \quad (2.1)$$

where $L = P(O|\lambda)$ is the likelihood of the model λ given the observed dataset O , n_p is the free parameter required for the modelling method, and N is the number of observation data used in fitting the model. However, the BIC score method requires multiple trials of the modelling process to determine L and n_p , and hence to find the optimal number of clusters / states which gives the lowest BIC score.

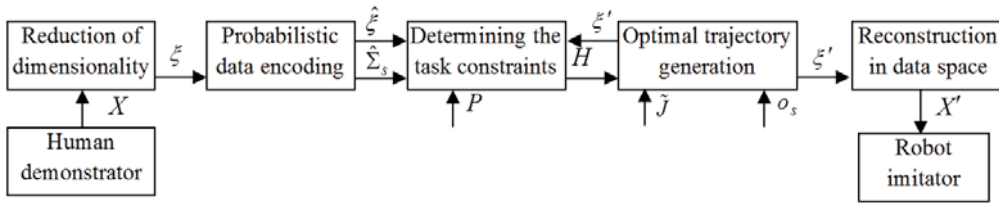


Figure 2-7 Information flow for robotic learning by demonstration described by Calinon et al. [61].

2.3.1.1 Hidden Markov Model

Hidden Markov Model has been widely used in data processing. HMM which is derived from Markov chain process assumes the current state z_k of a system only depends on its prior state z_{k-1} . A HMM is a double stochastic process, and the stochastic process is unobservable and only the emission is observable. The observation can be in the form of discrete data, or a probability density function (PDF). HMM is usually denoted by λ , $\lambda = \{\pi, \mathbf{A}, \mathbf{B}\}$ for discrete observations, where π is prior probability which states the initial probability of being in a certain state at time $t=0$. \mathbf{A} is a $n \times n$ matrix of transition probabilities. \mathbf{B} is a $n \times m$ observation matrix which states the probability of observing a particular state z_k .

HMM for continuous observation with Gaussian distribution is denoted as $\lambda = \{\pi, \mathbf{A}, \mathbf{\Gamma}, \mathbf{\mu}, \mathbf{\Sigma}\}$, where $\mathbf{\Gamma}$ is the mixture coefficient for every probability density function describing the Gaussian distribution, $\mathbf{\mu}$ is the mean vector, $\mathbf{\Sigma}$ is the covariance matrix.

The observation sequence O , HMM model λ , the probability of the observed sequence $P(O|\lambda)$ are the three key components in the HMM applications. By knowing or constraining the other two, the remaining components can be found. L.R. Rabiner et al. [71, 72] had systematically described the three problems and solutions formed by the components:

1. The probability of the observation sequence $P(O|\lambda)$ can be found by giving observation sequence O and HMM model λ ;

2. The optional state transition sequence S can be found by giving observation sequence O and HMM model λ ; and
3. The parameters in HMM model λ that gives maximum $P(O|\lambda)$ with given observation sequences O .

The above three problems can be solved by forward-backward algorithm, Viterbi algorithm and Expectation Maximization algorithm, respectively [71]. Forward-backward algorithm is applied to estimate $P(O|\lambda)$ in addressing the first problem with the assumption that the initial probability and observation probability and transition probability are known. The observation probability $P(O|\lambda)$ is expressed as

$$\begin{aligned} P(O|\lambda) &= \sum_I P(O|Z, \lambda) P(Z|\lambda) \\ &= \sum_{i_1, i_2, \dots, i_m} \pi_{i_1} b_{i_1}(O_1) a_{i_1 i_2} b_{i_2}(O_2) \dots a_{i_{m-1} i_m} b_{i_m}(O_m) \end{aligned} \quad (2.2)$$

Forward-backward algorithm consists of a forward algorithm and a backward algorithm. The forward algorithm computes the joint distribution probability $P(z_k, O_{1:k})$, $\forall k = 1 \dots m$, and the backward algorithm computes conditional distribution probability $P(O_{k+1:n} | z_k)$, $\forall k = 1 \dots m$. $P(z_k | O)$ is proportional to $P(z_k, O_{1:k})$, and it can be written as

$$P(z_k, O) = P(z_k, O_{1:k}) P(O_{k+1:n} | z_k), \forall k = 1 \dots m, \quad (2.3)$$

where $P(z_k, O_{1:k})$ which is denoted by $\alpha_k(z_k)$ can be found using forward algorithm

$$\alpha_k(z_k) = \sum_{q_{k-1}=1}^n P(O_k | z_k) P(z_k | z_{k-1}) \alpha_{k-1}(z_{k-1}), \forall k = 2 \dots m, \quad (2.4)$$

$$\begin{aligned}
\alpha_1(z_i) &= P(z_{1i}, O_1) \\
&= P(O_1 | z_{1i}) \\
&= \pi_1 b_{1i}, \forall i = 1 \dots n, k = 1,
\end{aligned}$$

and $P(O_{k+1:n} | z_k)$ denoted by $\beta_k(z_k)$ is found using backward algorithm

$$\begin{aligned}
\beta_k(z_k) &= \sum_{z_{k+1}=1}^n \beta_{k+1}(z_{k+1}) P(O_{k+1} | z_{k+1}) P(z_{k+1} | z_k), \forall k = 1 \dots m-1. \\
\beta_m(z_m) &= 1.
\end{aligned} \tag{2.5}$$

Viterbi algorithm is used to solve the second problem to find the states that produce the maximum probability with the given observation,

$$\arg \max [P(z_i | O_k)]. \tag{2.6}$$

However, the problem in Equation (2.6) can be simplified by solving

$$\arg \max [P(z_i, O_k)], \forall i = 1 \dots n, \forall k = 1 \dots m, \tag{2.7}$$

since $P(z_i, O_k)$ and $P(z_i | O_{1:k})$ are proportional.

Expectation Maximization method has been applied to estimate the parameters in HMM model. Baum-Welch algorithm is one of the Expectation Maximization methods. With an initial model λ and estimated model $\bar{\lambda}$, Baum-Welch algorithm estimates the parameter $\bar{\lambda}$ to produce $P(O | \bar{\lambda}) > P(O | \lambda)$, the HMM parameter is estimated by using $\bar{\lambda}$ to replace λ iteratively until the algorithm converges. The selection of initial model λ is critical in obtaining optimal parameters to represent the HMM but there is no guarantee that global maximum can be reached by Baum-Welch algorithm. Researchers have also applied different methods in searching the optimal parameter, such as Tabu search [73, 74], Genetic Algorithm [75] and hybrid particle swarm optimization method [76]. In practice, initial model λ can be selected based on the assumption that the observation data follow certain distribution property such as Gaussian distribution.

With the state transition model \mathbf{B} , the HMM works with full transition model, left to right model and others. Figure 2-8 illustrates the different states transition patterns. In the implementation of HMM, only the corresponding b_{ij} indicated on the transition patterns has a probability value, the rest of the elements in the transition matrix \mathbf{B} are all zero. Full transition model allows transition to occur from any state to any other states or itself. There is no constraint on the transition between states. It can be applied to model the 'mental state' in motion intention recognition where the user could actually revisit the same 'mental state' for different tasks.

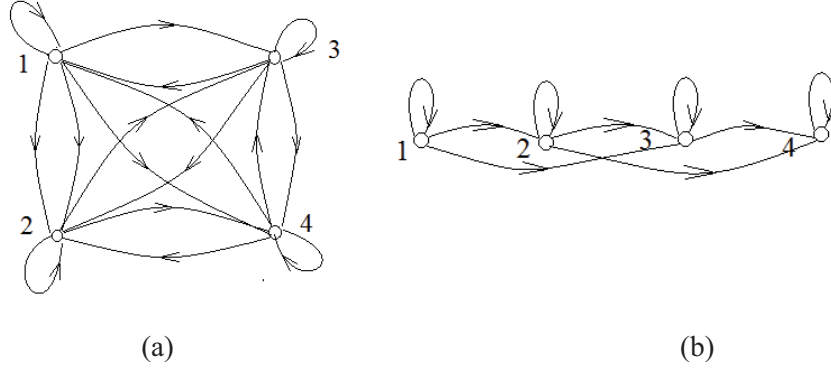


Figure 2-8 State transition patterns (a) full transition, (b) left to right transition.

2.3.1.2 Hidden Markov Model Approach

Calinon et al. [70] and Hundtofte et al. [68] applied HMMs to model the gestures of human. Aarno et al. [69] applied HMMs to model the motion trajectories of a user in picking / putting subjects. K-means method was applied to cluster the demonstration trajectories. A set of time series $\{\xi^x(t), \xi^\theta(t)\}$ clusters expressed in latent space was applied to train a fully connected continuous HMM with $K^x + K^\theta$ output variables. $\xi^x(t)$ and $\xi^\theta(t)$ is the hand path and joint angle of the robot, respectively, in Calinon's study [70]. The HMM is expressed by the parameters set $\lambda = \{\pi, \mathbf{A}, \mathbf{\Gamma}, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$

representing the initial states distribution, the states transition probabilities, the means of the output variables, and the standard deviations of the output variables, respectively. For each state, the output variables are described by multivariate Gaussians, i.e. $P(\xi_i^\theta) \sim N(\mu_i^\theta, \sigma_i^\theta)$ and $P(\xi_i^x) \sim N(\mu_i^x, \sigma_i^x)$. The transition probability $P(q(t) = j | q(t-1) = i)$ and the observation distribution $P(\xi(t) | q(t) = i)$ can be estimated by the Expectation Maximization algorithm.

The trained HMM can be used to recognize whether a new gesture is similar to the ones encoded in the model. For each HMM, Calinon et al. applied a forward algorithm to estimate the likelihood L that the observed data O could have been generated by the model λ , i.e. $\log(L) = \log(P(O | \lambda))$. A gesture is said to belong to a given model when the associated log likelihood is greater than a given threshold. In order to compare the prediction of two concurrent models, a minimal threshold for the difference across log-likelihoods of the two models needs to be set ($\Delta \log(L) > 100$ was set in [70]). Thus, for a gesture to be recognized by a given model, the voting model must be very confident (i.e. producing a high $\Delta \log(L)$), while other model predictions must be sufficiently low in comparison. However, the method of setting the threshold values for log likelihood and difference of log likelihood of concurrent models was not discussed in the paper.

After a gesture has been recognized, Calinon et al. [70] applied the following steps to reconstruct an optimal trajectory for the gesture:

1. The best sequence of states (according to the model's parameters $\{\pi, \mathbf{A}, \Gamma, \boldsymbol{\mu}, \boldsymbol{\Sigma}\}$) is extracted using the Viterbi algorithm;

2. A time-series of $K^x + K^\theta$ variables $\{\xi^{''x}(t), \xi^{''\theta}(t)\}$ is generated by computing the mean values μ of the Gaussian distribution of each output variable at each state;
3. The time series in step 2 is reduced to a set of key-points $\{\xi^{''x}(t), \xi^{''\theta}(t)\}$ in between each state transition;
4. A set of output variables $\{\xi^{'x}(t), \xi^{'\theta}(t)\}$ is generated by interpolating between these key-points and normalizing in time; and
5. Dimensionality of $\{\xi^{'x}(t), \xi^{'\theta}(t)\}$ is recovered into the robot's workspace for execution.

2.3.1.3 Gaussian Mixture Approach

There are two components in Gaussian Mixture approach: Gaussian Mixture Model (GMM) and Gaussian Mixture Regression (GMR). Gaussian Mixture approach has been used in humanoid robots to learn and execute some tasks, such as grasping an object [59, 61, 77].

A. Gaussian Mixture Model

Gaussian Mixture Model is a linear superposition of Gaussian components K_p , defined by probability density function

$$P(\mathbf{x}) = \sum_{k_p=1}^{K_p} P(k_p) P(\mathbf{x} | k_p), \quad (2.8)$$

where $P(k_p) = \pi_{k_p}$ is the prior probability.

$$P(\mathbf{x}_i | k_p) = N(\mathbf{x}_i; \boldsymbol{\mu}_{k_p}, \boldsymbol{\Sigma}_{k_p})$$

$$= \frac{1}{\sqrt{(2\pi)^d |\boldsymbol{\Sigma}_{k_p}|}} e^{-\frac{1}{2}((\mathbf{x}_i - \boldsymbol{\mu}_{k_p})^T \boldsymbol{\Sigma}_{k_p}^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_{k_p}))}$$

is the conditional probability density

functions for component k_p . $P(\mathbf{x})$ is the probability that the trajectory data point \mathbf{x}_i is constructed by the model.

The parameters of the GMM are expressed as $\{\boldsymbol{\pi}_{k_p}, \boldsymbol{\mu}_{k_p}, \boldsymbol{\Sigma}_{k_p}, E_{k_p}\}$, where $\boldsymbol{\pi}_{k_p}$ is the prior probability, $\boldsymbol{\mu}_{k_p}$ is the mean vector, $\boldsymbol{\Sigma}_{k_p}$ is the covariance matrix and

$E_{k_p} = \sum_{i=1}^N p(k_p | \mathbf{x}_i)$ is the cumulated posterior probability. The number of the components K_p is obtained by K-means clustering method [61]. The trajectory data \mathbf{x}_i contains the temporal and the spatial information. It is expressed as $\mathbf{x}^N = \{\mathbf{x}_{t,i}, \mathbf{x}_{s,i}\}^N$, and hence the mean vector is expressed as $\boldsymbol{\mu}_{k_p} = \{\boldsymbol{\mu}_{t,k_p}, \boldsymbol{\mu}_{s,k_p}\}$,

the covariance matrix is expressed as $\boldsymbol{\Sigma}_{k_p} = \begin{pmatrix} \boldsymbol{\Sigma}_{tt,k_p} & \boldsymbol{\Sigma}_{ts,k_p} \\ \boldsymbol{\Sigma}_{st,k_p} & \boldsymbol{\Sigma}_{ss,k_p} \end{pmatrix}$.

The GMM's parameters $\{\boldsymbol{\pi}_{k_p}, \boldsymbol{\mu}_{k_p}, \boldsymbol{\Sigma}_{k_p}, E_{k_p}\}$ can be estimated by Expectation Maximization algorithm (EM) [61] with the demonstration trajectories as the training data.

B. Gaussian Mixture Regression

Gaussian Mixture Regression (GMR) can be applied to reconstruct a trajectory represented by the GMM. The regression method estimates the conditional expectation of $\hat{\mathbf{x}}_s$ with given \mathbf{x}_t , and hence the entire trajectory can be reconstructed with motion's major characteristics preserved. For the k_p^{th} component at given time \mathbf{x}_t , the expected distribution of \mathbf{x}_{s,k_p} is

$$P(\mathbf{x}_{s,k_p} | \mathbf{x}_{t,k_p}) = \mathcal{N}(\mathbf{x}_{s,k_p}; \hat{\mathbf{x}}_{s,k_p}, \hat{\Sigma}_{ss,k_p}), \quad (2.9)$$

where \mathbf{x}_{s,k_p} and Σ_{ss,k_p} is the conditional expected value of the component k_p

and expected covariance. They are expressed as

$$\mathbf{x}_{s,k_p} = \boldsymbol{\mu}_{s,k_p} + \Sigma_{st,k_p} (\Sigma_{tt,k_p})^{-1} (\mathbf{x}_t - \boldsymbol{\mu}_{t,k_p}), \quad (2.10)$$

$$\Sigma_{ss,k_p} = \Sigma_{ss,k_p} + \Sigma_{st,k_p} (\Sigma_{tt,k_p})^{-1} \Sigma_{ts,k_p}. \quad (2.11)$$

\mathbf{x}_{s,k_p} and Σ_{ss,k_p} are combined based on the probability that the component k_p

for the given time \mathbf{x}_t :

$$P(\mathbf{x}_s | \mathbf{x}_t) = \sum_{k_p=1}^{K_p} \beta_{k_p} \mathcal{N}(\mathbf{x}_s; \hat{\mathbf{x}}_{s,k_p}, \hat{\Sigma}_{ss,k_p}), \quad (2.12)$$

$$\text{where } \beta_{k_p} = \frac{P(k_p)P(\mathbf{x}_t | k_p)}{\sum_{i=1}^{K_p} P(i)P(\mathbf{x}_t | i)} = \frac{\pi_{k_p} \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{t,k_p}, \Sigma_{tt,k_p})}{\sum_{i=1}^{K_p} \pi_i \mathcal{N}(\mathbf{x}_t; \boldsymbol{\mu}_{t,i}, \Sigma_{tt,i})}.$$

An estimation of the conditional expectation of \mathbf{x}_s at the given time \mathbf{x}_t for component k_p^{th} in the mixture model is

$$\mathbf{x}_s = \sum_{k_p=1}^{K_p} \beta_{k_p} \mathbf{x}_{s,k_p}, \Sigma_{ss} = \sum_{k_p=1}^{K_p} \beta_{k_p}^2 \Sigma_{ss,k_p}. \quad (2.13)$$

The generalized form of the motion trajectory can then be expressed as

$$\hat{\mathbf{x}} = \{\hat{\mathbf{x}}_t, \hat{\mathbf{x}}_s\}.$$

C. Motion Imitation

Billard et al. [78] proposed a general formalism for evaluating the reconstruction of a task

$$H = \frac{1}{K} \sum_{i=1}^K w_i \bullet H_i(x_i, x_i'), \quad (2.14)$$

where x_i and x_i' are the candidate trajectory and the generalized trajectory, respectively, and w is the weight vector.

The generic similarity measure H takes into account the variations of constraints and the dependences across the variables which have been learned over time. The matrix is continuous, positive, and can be estimated at any point along the trajectory. In the latent space, let $\{\hat{\xi}_s^\theta, \hat{\xi}_s^x, \hat{\xi}_s^y\}$ be the vector of generalized joint angle trajectories, the generalized hands paths, and the generalized hands-object distance vectors extracted from the demonstrations, respectively. Let $\{\xi_s^\theta, \xi_s^x, \xi_s^y\}$ be the candidate trajectories for constructing the motion. The metric of imitation performance H is given by

$$H = (\xi_s^\theta - \hat{\xi}_s^\theta)^T W^\theta (\xi_s^\theta - \hat{\xi}_s^\theta) + (\xi_s^x - \hat{\xi}_s^x)^T W^x (\xi_s^x - \hat{\xi}_s^x) + (\xi_s^y - \hat{\xi}_s^y)^T W^y (\xi_s^y - \hat{\xi}_s^y), \quad (2.15)$$

where T is the number of observations.

Calinon et al. [61] further proposed to transfer the imitation performance H from position domain into velocity domain by setting

$$\begin{aligned} r_{\theta,i,j} &= \hat{\xi}_{s,i,j}^\theta - \xi_{s,i,j-1}^\theta \\ r_{x,i,j} &= \hat{\xi}_{s,i,j}^x - \xi_{s,i,j-1}^x, \quad i=1\dots n_p, j=1\dots T, \\ r_{y,i,j} &= \hat{\xi}_{s,i,j}^y - \xi_{s,i,j-1}^y \end{aligned} \quad (2.16)$$

and

$$H = (\dot{\xi}_s^\theta - c_1)^T W^\theta (\dot{\xi}_s^\theta - r_1) + (\dot{\xi}_s^x - r_2)^T W^x (\dot{\xi}_s^x - r_2) + (\dot{\xi}_s^y - r_2)^T W^y (\dot{\xi}_s^y - r_3). \quad (2.17)$$

The optimal trajectory of the robot with object constraint can be obtained by finding $\{\xi_s^\theta, \xi_s^x, \xi_s^y\}$ with minimum H . Calinon et al. [61] solved it using Lagrange optimization method.

2.3.2 Neural Networks Methods

Herman et al. [54] applied Recurrent Neural Networks (RNNs) to learn tying surgical knots in MIS. Long Short-Term Memory (LSTM) approach was

used to overcome the problem with standard gradient descent techniques in RNNs. The specialized linear memory cells in LSTM can maintain their activation indefinitely. Enforce Subpopulation algorithm (ESP) was applied to train the LSTM networks. Figure 2-9 showed the application of ESP to train LSTM with given knot tying trajectories demonstrated by the surgeon.

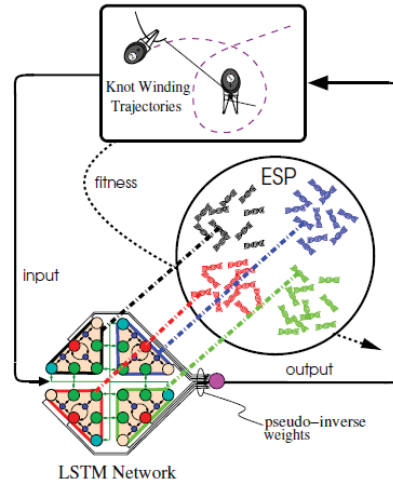


Figure 2-9 Enforced Sub Populations (ESP) neuroevolution method. The Long Short-Term Memory (LSTM) network architecture (shown with four memory cells), and the pseudo inverse method to compute the output weights. When a network is evaluated, it is first presented with the training set to produce a sequence on network activation vectors that are used to compute the output weights. Then the training set is presented again, but now the activation also passes through the new connections to produce outputs. The error between the outputs and the targets is used by ESP as a fitness measure to be minimized (source: [54]).

2.4 User Intention Recognition for Human Robot Collaboration

With the skills learnt by the robot, the next step is to recognize the intention of the users / residents / novice surgeons during training, so that, the robot can react according to the skills it has learnt to provide assistance. Research on intention recognition has been going on for more than three decades. It has

been applied in understanding the intention in language, context of story, motion, etc. In this study, the objective is to understand the intention of user from their motion trajectory.

Intention recognition problem can be classified as intended and keyhole [79] problem. In the intended case, the observed agent wants his / her intentions to be identified and intentionally gives signals to be sensed by the other observing agent. For example, in the case of language understanding where the speaker wants to convey his / her intentions and captured by the other observing agent. In the keyhole case, the observed agent either does not intend for his / her intentions to be identified or does not care. The observed agent focuses on his / her own activities. The observed agent may provide only partial observability to the observing agent. This might be relevant to assistance systems that provide unsolicited guidance. In our robot-assisted surgical training system, the user / novice surgeon's purpose is to perform and complete a specific surgical task. The user / novice surgeon does not purposely communicate his / her intention to the robotic surgical trainer. Therefore, the robot needs the capability of intention recognition with keyhole.

There are two approaches for intention recognition in the literature: logic-based approach [79] and probabilistic approach [80-84]. Logic-based intention recognition constructs the problem in the form of plan libraries [85, 86], situation calculus [87], event calculus [88], causal Bayesian nets [89, 90] or multi-context logical theories with bridge rules [91] etc., and using abductive [85, 86, 88, 91], Bayesian [89, 90] or probabilistic [87] approaches for reasoning. Logic-based intention recognition methods are more suitable for high level activities' intention recognition which the events in an activity can

be identified easily, such as context of language [88, 91], story understanding, tactical plan recognition (enemy's plans) [86], terrorist intention recognition [85] and elder care [89].

Charniak and Goldman suggested that the problem of intention recognition is largely a problem of inference under conditions of uncertainty [92]. Probabilistic reasoning approaches have profound advantages in solving problems where uncertainties are intrinsic [93]. Researchers [80-84] have been devoting efforts in probabilistic-based intention recognition for low level motion intent recognition, such a HMM [41, 80, 81] probabilistic state machine [82, 83] and dynamic Bayesian Network [84, 94]. In the following subsections, the intention recognition method using probabilistic approach is reviewed.

2.4.1 Hidden Markov Model

HMM has been widely studied for intention recognition. He et al. [95] applied a double layered HMM to recognize the intention of a driver during driving. Hou et al. [96] applied continuous HMM to recognize the driver's intention to change lane. Aaron et al. [69, 80, 97] applied K-means, Support Vector Machine (SVM), HMM to automatically extract the motion primitive, and intention of operator. A virtual fixture was constructed based on the motion primitive and intention of operator for assistance in teleoperation tasks. Zhang et al. [98] applied HMM to recognize potential intrusion activities for cyber security. Aaron et al. [69, 80, 97] applied a Layered HMM to observe the positional trajectory generated by user, and extract the intention of the user.

In the Layered HMM, each layer has its own functionality. In a trajectory tracking task, the observations are acquired from classification of the previous

layer, except the lowest layer where the observations are acquired directly from measurements of the observed process. Figure 2-10 illustrated the Layered HMM by Aaron et al. [80, 97]. Figure 2-11 illustrated a two level layered HMM. At Level 1, a single HMM is used to model the task, where each state in the HMM corresponds to a subtask. At Level 2, there is a HMM for each possible gesteme that may occur during execution of the task. The observation sequence for the Level 2 HMMs is generated from the motion direction of the trajectory recorded during task operation. The index of the HMM with highest likelihood among the various HMMs at Level 2 is then taken to be the observation symbol for the Level 1 HMM. The Level 1 HMM is then used to compute the probability of a certain state as a function of time given the observation sequence produced by the HMMs at Level 2. Since each state in the Level 1 HMM corresponds to a mental stage of the operation task, this information can be used to understand the operator's intention. The winning HMM at Level 2 which is the one with the highest likelihood will be chosen. An observation symbol corresponding to this gesteme is generated for the Level 1 HMM. The alternative would be to use the complete probability distribution and have the HMMs at Level 2 act as a probability estimator for the Level 1 HMM. However, Oliver et al. [99] found that using the complete distribution does not give any apparent advantage over the simpler winner takes all models.

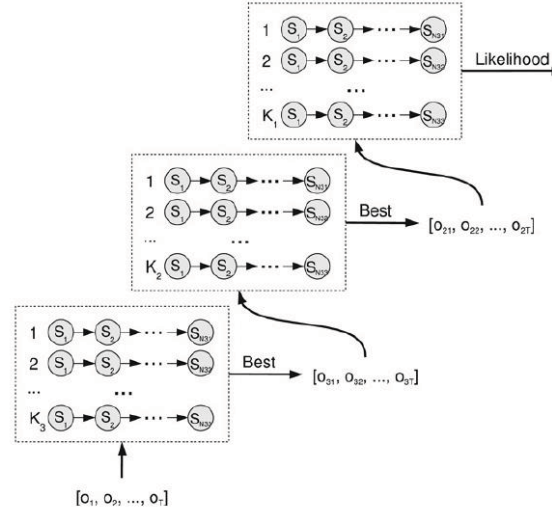


Figure 2-10 Structure of layered hidden Markov Model proposed in [80].

In this layered HMM, HMMs at different layer was assigned with different objectives. For example, the HMMs in layer 2 are classifiers by classifying the trajectory signals into gestemes [80, 97]. The HMMs in layer 1 recognize the intention related to mental state.

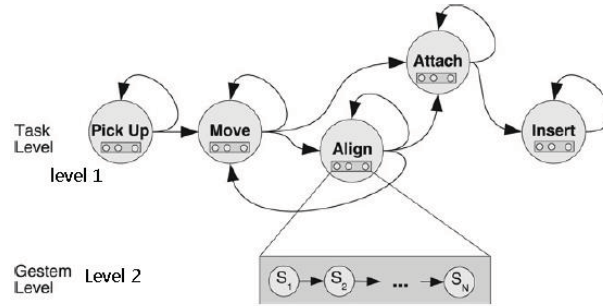


Figure 2-11 Two-level layered hidden Markov model implemented by D. Aaron et al. [80], modelling gestemes at level 2 and task a level 1. Level 2 acquires signal from trajectory [80].

Feature representation is an important factor in the application of HMM. There are two basic approaches to determine the input presentation for recognition problem [100]. One approach is using the raw data as the input. It is named as 'template' in [100]. 'Template' is one of the major features in gesture recognition. The 'template' can be formed by the coordinate points of the trajectory path. Yang et al. [100] noted that one of the drawbacks of using

'template' data as feature is the low robustness within a given class. The other approach is using the extracted important features, such as frequency information. The advantage of using features from frequency domain is the stability of recognition. Feature for training HMM will be discussed in Section 2.5.1.

2.4.2 Probabilistic State Machine

Probabilistic state machine approach is another method applied by researchers for intention recognition from the motion. A general form of probability state machine is illustrated in Figure 2-12. It is described by a tuple

$$A = \{Z_A, \Omega, \delta_A, I_A, F_A, P_A\}, \quad (2.18)$$

where Z_A is a finite set of states.

Ω is the alphabet / a set of actions,

$\delta_A \subseteq Z_A \times \Sigma \times Z_A$ is a set of transitions,

$I_A : Z_A \rightarrow \mathbb{R}^+$ is the initial-state probabilities,

$P_A : \delta_A \rightarrow \mathbb{R}^+$ is the transition probabilities,

$F_A : Z_A \rightarrow \mathbb{R}^+$ is the final -state probabilities,

The state machine system meets the condition:

$$\sum_{q \in Q_A} I_A(z) = 1 \text{ and } \forall z \in Q_A, F_A(z) + \sum_{a \in \Sigma, z' \in Z_A} P_A(z, a, z') = 1.$$

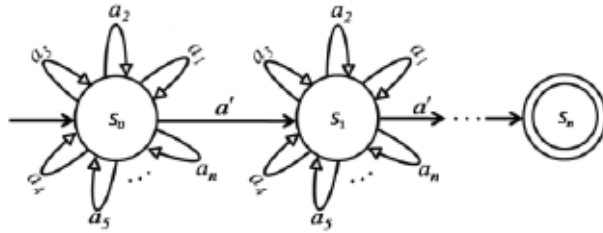


Figure 2-12 General representation of probabilistic state machines in transition [83].

Wais et al. [83] applied the probabilistic state machine approach to enable the robot to recognize the intention of human collaborator when the human and robot are collaborating to complete a specific task. They divided the intentions into explicitly communicated intentions and implicitly communicated intentions which are similar to the intended or keyhole intention recognition defined by Cohen et al. [79]. Examples with the two different communicated intentions were studied in their work [83]. They studied an explicitly communicated intention example that a human user pointed an object to the robot, and then robot picked up the object and placed the object at a specified location. In implicitly communicated intention example, the robot observes human arm movement, and also the changes of objects numbers, the intention of piling and unpling is represented by probabilistic state machines. Figures 2-13 (a) and (b) show the probabilistic state machine.

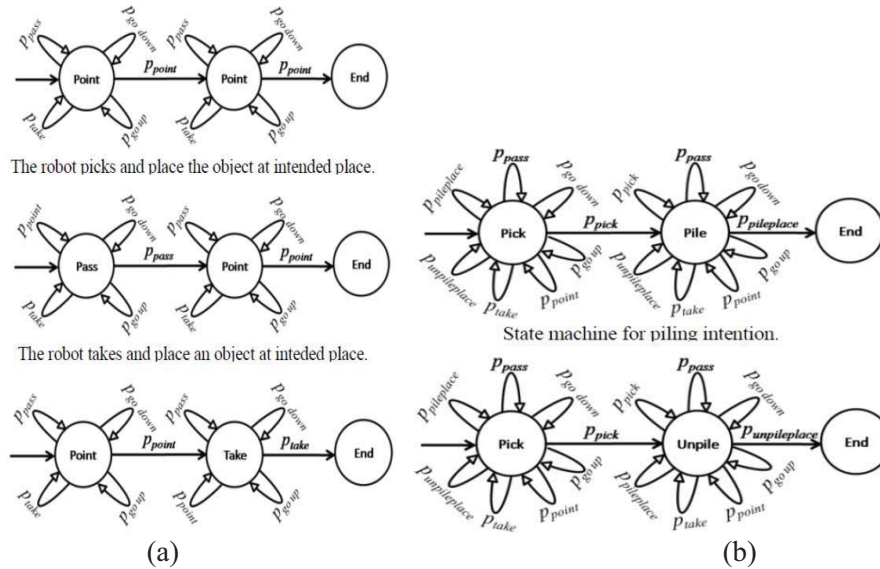


Figure 2-13 Probabilistic state machines for piling and unpling intentions in [83]. (a) State machine for explicitly communicated human intentions (picking and placing an object). (b) Implicitly communicated human intentions.

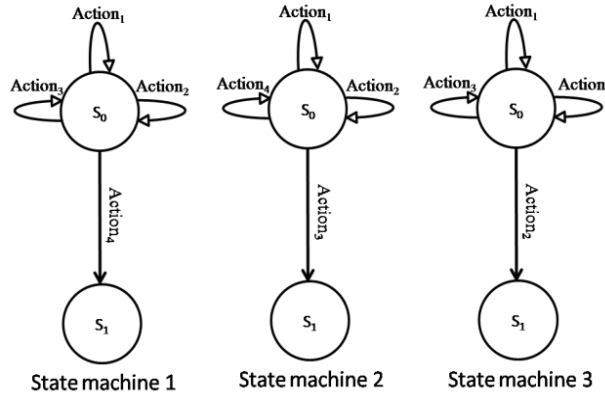


Figure 2-14 Intention recognition algorithm presented in [83].

Wais et al. [83] proposed an algorithm to detect the intention of the human user from observation data. The intention recognition algorithm represents each state machine as a unique explicitly / implicitly communicated human intention. The weights of the state machine are the same. When an observation is made, and feature information is extracted, the weights of the state machines are updated. The weights are normalized so that they add up to unity. The state machine for which the observation is most probable gets higher weight as compared to the other state machines. If an observation is equally probable for some state machines then those state machines get the same normalized weight. After an observation, if there is no human action observed or an irrelevant human action is observed, no state transition occur in any state machine. If a relevant human action is observed, then the current active states of all the state machines will be checked. If the observation has the highest probability for the current active state then the state transition will occur in that state machine. If the observation is highly probable for more than one state machine (current active state) then the state transition will occur in more than one state machine. In other state machines, no state transition will occur. This is explained in Figure 2-14.

Changing of intentions is taken care of by making transition in most the probable state machines. For example, if the human has an intention and performs an action then the concerned state machine (intention) gets high weight, and a transition occurs only in that state machine. If the human changes his / her intention then the new action sequence can be evaluated with the related state machine and the changed intention can be easily recognized. The strategy in handling of changing intention may cause a recognition fault. For example, a user changes from intention I1 into I2, and an action performed in the action sequence of I2 is required for I1, then the false intention will only be recognized if I1 reaches its final state and has a high weight. If the end state of a state machine is reached and the state machine has the highest weight then that intention is recognized and state machines are reinitialized. If the end state is reached but the weight is not the highest then the re-initialization is performed without intention recognition [83].

Nguyen et al. [101] applied probabilistic state machine approach to recognize the intention of a computer game player, and hence to derive a policy to select an action for a computer collaborator's execution in a game task. The task was divided into subtasks manually. In his research, the subtask is defined by catching one counter party of the player. At each time instance, the player is assumed to be likely to continue on the subtask that he or she is currently pursuing. However, there is a small probability that the player may decide to switch subtasks. The Probability that Player intends to continuously pursuing his / her chosen sub-goal is 80%, and the probability of switching to other sub-tasks is 20%, as illustrated in Figure 2-15. The transition probability distributions of the nodes need not be homogeneous, as the human player

could be more interested in solving some specific subtask right after another subtask. For example, if the counter party needs to be captured in a particular order, this constraint can be encoded in the state machine. The model also allows the human to switch back and forth from one subtask to another during the course of the game, modelling change of mind.

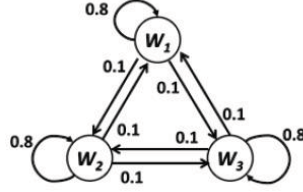


Figure 2-15 A probabilistic state machine that models the transitions between subtasks in [101].

Belief function has been applied in Nguyen's study. The belief at time t is denoted by $B_t(w_i | \theta_t)$, where w_i is the i^{th} subtask and θ_t is the game history. The belief function is the conditional probability of that the player is performing subtask w_i . The belief update operator takes $B_{t-1}(w_i | \theta_{t-1})$ as input and carries out two updating steps:

1. The next subtask belief distribution is obtained by the following equation

$$B_t(w_i | \theta_{t-1}) = \sum_j T(w_j \rightarrow w_i) B_{t-1}(w_j | \theta_{t-1}), \quad (2.19)$$

where $T(w_j \rightarrow w_i)$ is the switching probability from subtask j to subtask i .

2. The posterior belief distribution is calculated by using Bayesian update after observing the players action a and subtask state $s_{i,t}$ at time t , as follows

$$B_t(w_i | a_t = a, s_t, \theta_{t-1}) = \alpha \cdot B_t(w_i | \theta_{t-1}) \cdot P(\alpha_t = a | w_i, s_{i,t}), \quad (2.20)$$

where α is a constant for normalization. Current human action a and current state and game history θ_{t-1} forms the game history θ_t at time t .

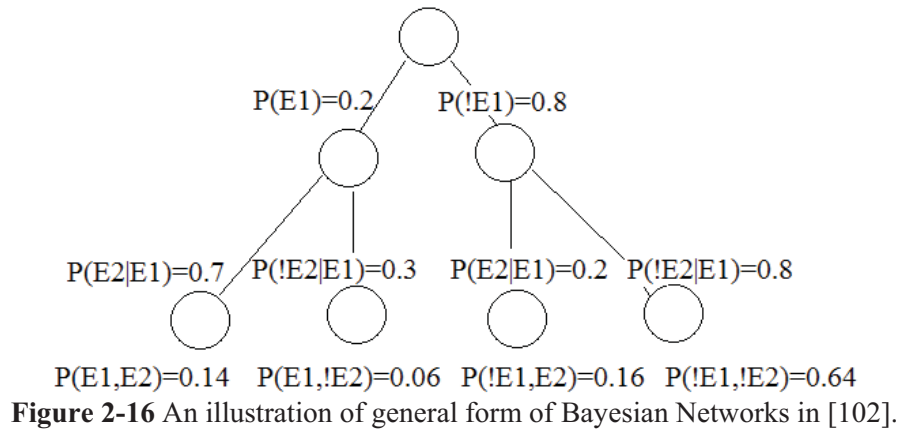
With a belief distribution on the player targeted subtasks as well as knowledge to act collaboratively and optimally on each of the subtasks, the computer agent chooses the action that maximizes its expected reward to achieve the purpose of collaboration,

$$a^* = \arg \max_a \left\{ \sum_i B_t(w_i | \theta_t) Q_i(s_t^i, a) \right\}, \quad (2.21)$$

where Q_i is the expected long term reward of an action a when executed in states s .

2.4.3 Dynamic Bayesian Networks Approach

A Bayesian Network (BN) is a directed acyclic graph encoding assumptions of conditional independence. To define a BN, as shown in Figure 2-16, structure of the network, the conditional probability distribution and the prior probability distribution of the top nodes are required. The nodes in BN represent stochastic variables; whereas the lines connected two nodes represent causal dependence. Associated with each node is a specification of the distribution of its variable conditioned on its predecessors in the graph. Such a network defines a joint conditional probability distribution (CPD); that is the probability of an assignment to the stochastic variables is given by the product of the probabilities of each node conditioned on the value of its predecessors according to the assignment.



In the application of BNs for intention recognition, the intentions are represented by top nodes. The result of the intention is a desired state. Actions and changes in states that follow from these intentions and desired states are represented by nodes below which are connected causally to intention nodes. Bottom nodes are usually connected to the observable signals from sensors. The prior probability distribution of the intention nodes reflects the intending agent's mental state. Here, only the values of the observed nodes are used to compute the posterior probability distribution for intention nodes.

Intention recognition is always a temporal dynamics process. Dynamic Bayesian Network (DBN) has been applied to solve the intention recognition problem. DBN is the architecture of Bayesian networks (BN) for representing the evolution of variables over time. It consists of a sequence of time slices where each time slice contains a set of variables representing the state at the current time. A DBN is formed by three information: the prior distribution over the variables, the transition model (from one time slice to the next), and the conditional probability distribution.

Tahboub et al. [84] applied Dynamic Bayesian Network to recognize the intention from user for the human-machine interaction. They applied DBN to

model the intention recognition process “intention–activity–state” which is proposed by Heinze [103]. They proposed a time delay method to eliminate the cycles that arise due to the feedback of sensed states to the intention and action nodes which violate the definition of DBN. Figure 2-18 illustrates the changes made on the process flow. The cycles are eliminated by feeding back the sensed states from a previous time slice instead of the current one.

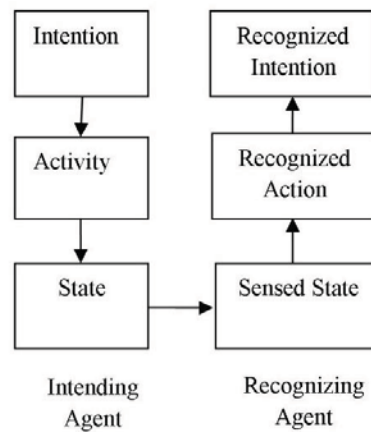


Figure 2-17 Work path of intention recognition in [103].

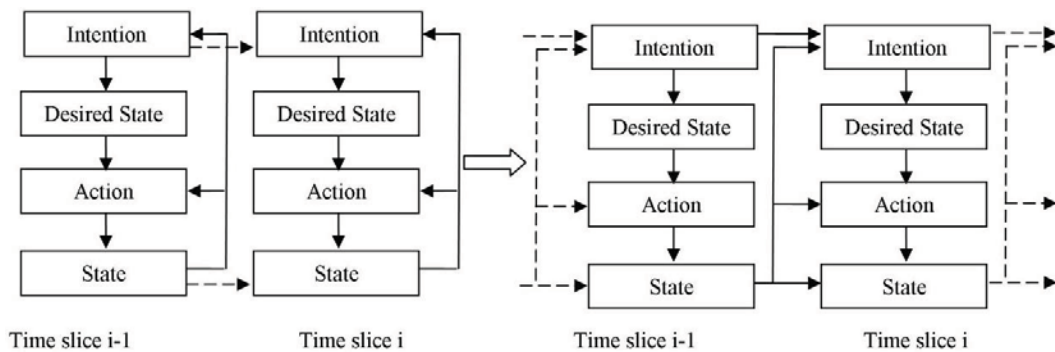


Figure 2-18 Human’s intention-action-state flow and DBN corresponding to human intention recognition model with time-delay presented in [84].

2.5 Performance Evaluation Methods

An objective evaluation method is required to evaluate the effectiveness of a surgical training system. Such evaluation is achieved by measuring the performance of user / surgeon’s surgical skills before and after being trained

by the surgical training system. Surgical skills are traditionally evaluated manually. The common practice in such evaluation is to conduct a scoring process by several experts based on a performance matrix.

The evaluation of intraoperative laparoscopic surgical skill has been well established. Vassiliou et al. [104] presented a global assessment tool to rate the performance of the surgeon in laparoscopic cholecystectomy. He summarized the assessment criteria for intraoperative laparoscopic skills. It consists of a 5-item global rating scale and a task-specific check list as shown in Table 2-1 and Table 2-2, respectively. Five aspects were considered during the rating process, such as depth perceptions, bimanual dexterity, efficiency, tissue handling and autonomy. Table 2-2 shows a task-specific checklist for laparoscopic cholecystectomy skill assessment [104]. The presented method is a summary of expertise in the field. It gives a very detailed view from the medical staff's perspective. However, such evaluation results are subjected to bias, experience and perspective of the evaluator. Although the effect can be minimized by increasing the number of evaluators, it increases the cost of evaluation dramatically.

The evaluation results from manual evaluation are hardly free from subjective bias carried by the evaluator. Other researchers also tried to develop objective methods that can assess the performance of a surgeon during the surgery or exercise on the surgical simulator. Motion trajectory [105-108], force torque signature [109-111] and video [107] were taken as the study objects. HMM [107-109, 111], Fuzzy logic [105, 112] and LDA [106] have been used in various studies.

Table 2-1 Five-item global rating scale described by Vassiliou et al. [104].

Bimanual		Efficiency	Tissue handling	Autonomy	others
Depth perception	dexterity				
No of Overshooting	Distance difference to expert's liver fixation point	Time to fix the liver	Tissue tearing	Completeness	Perforation of GB/Spillage of Gallstones/un desired injuries
Wide swing angles	Deformation of Gallbladder	Time to expose the ducts	Tissue trauma / bleeding	Guidance	
Time taken to correct the overshooting	No. of attempts for Clipping	Time to complete clipping	Grasper slip		Hesitation
	No. of attempts for cutting	Time to complete cutting	Force torque signature		

Table 2-2 Task-specific checklist presented in [104]: dissection of the gallbladder from the liver bed.

Tasks	Done (1 point)	Not Done (0 point)
1. Uses cautery only when all conducting areas are in field of view		
2. Has good control of the instrument, minimizes recoil		
3. Grasps gallbladder near clips to begin dissection		
4. Readjusts tension on gallbladder to optimize exposure		
5. Avoids dissecting into liver causing undue bleeding		
6. Avoids perforation of the gallbladder		
7. Avoids spillage of gallstones		
8. Maximizes useful dissection in 1 area before changing approach		
9. Performs dissection in appropriate plane the majority of the time		
10. Obviates the need for surgeon takeover		
Total (10)		

Chuan et al. [105, 112] developed a fuzzy logic-based framework to evaluate the performance of a surgeon. Motion information from a virtual simulator and motion information acquired from tracking system on the surgical instruments were used for evaluation. The framework looked into the basic assessment criteria in surgery, such as total time, total path length, average speed, instant speed and motion area radius.

HMM is another popular instrument that has been applied to compare and evaluate the performance of surgeons. Reiley et al. [59] applied GMM method to learn suture tying task from expert surgeons' demonstration, and HMM method to classify the motion trajectory performed by other surgeons into three levels based on log likelihood distance. This method provides an objective evaluation of the subjects. However, it is unable to give detailed evaluation as described by Vassiliou et al. [104]. The three levels indicated the general similarity between the test subject and a generic mode.

2.5.1 Features for Evaluation Methods

Different features have been used in evaluating the surgeon's performance, including velocity, frequency, force / torque. Lin et al. [106] applied the LDA method to distinguish the proficiency level of expert surgeon and novice surgeon in using a surgical simulator for training. Motion data of user's upper body were tracked in the study. Seven features were used in the study: left and right shoulder average angular speed; left and right shoulder angular speed cumulative distribution function (CDF); left and right shoulder angular speed frequency, and right-shoulder-used ratio; and the average power spectrum density (PSD) of joints' angular speed which can be extracted to determine the right-shoulder-used ratio. It is defined as

$$\rho = PSD(shoulder) / PSD(elbow) + PSD(wrist). \quad (2.22)$$

Features used for HMM method varies, such as force / torque [111], bag of words from the surgical training video [107], and spatial trajectory.

Rosen et al. [111] took the force / torque applied on the instrument as observation features for HMM training. In his study, each observation was

composed of seven components vector of data $\mathbf{f} = (f_x \ f_y \ f_z \ \tau_x \ \tau_y \ \tau_z \ f_g)$. Force / torque were continuous stream of data distributed normally, each HMM state could be defined by seven normal distributions functions chartered by a mean and a standard deviation $(N_i(\mu, \Sigma) \ i=1...7)$. The 7 element vector was combined into a joint multivariable distribution function $f(\mathbf{f})$. It was done by using Equation (2.23):

$$f(\mathbf{f}) = \frac{1}{(\sqrt{2\pi})^N |\Sigma|^{1/2}} e^{-\frac{1}{2}(\mathbf{f}-\mu)\Sigma^{-1}(\mathbf{f}-\mu)}, \quad (2.23)$$

where \mathbf{f} is the Force / torque observation vector; μ is the mean vector; Σ is the covariance matrix, and N is the observation vector size.

Megali et al. [108] defined the observation feature as the cluster center of motion trajectory in the frequency domain. Qiang et al. [107] took the “bag of words” approach for feature extraction from the videos taken from surgery. The spatiotemporal interest point detector [113] was applied to obtain the histogram-of gradient (HoG) features. K-means was then used to build a codebook for the descriptors of the interest points. Finally, the codebook was used to obtain a histogram of interest points for each frame, and thus each video is represented as a sequence of histograms. This representation could better capture the temporal information of the video.

2.5.2 Evaluation Methods

Towards an objective evaluation of surgeon's performance, several methods has been studied by the researchers, such as HMM [107-109, 111], Fuzzy logic [112] and LDA [106] methods. In the following section, these methods are reviewed.

2.5.2.1 Hidden Markov Model for Evaluation

The concept of using HMM method is motivated by the 3 problems that HMM can solve stated in [71, 72]. Feature extracted from expert surgeon's execution were applied to train a HMM λ_s . Probability of the observation sequence from the novice surgeon $P(O|\lambda_s)$ was taken as a basic measurement of similarity between the novice surgeon and the expert surgeon in the scope of the observation feature.

In using HMM methods to evaluate the surgeon's performance, it is essential to train the model stable in representing one surgeon or several surgeon skills. Megali et al. [108] verified the expert surgeon's model by minimizing a distance function of two expert surgeon's model λ_{s1} and λ_{s2} , i.e.

$\min_{N, obs, Wl} D_{sm}(\lambda_{s1}, \lambda_{s2})$. He proposed the distance function as

$$D_{sm}(\lambda_{s1}, \lambda_{s2}) = \frac{1}{T_{O^{s2}}} [\log P(O^{s2} | \lambda_{s1}) - \log P(O^{s2} | \lambda_{s2})], \quad (2.24)$$

where $T_{O^{s2}}$ is the duration of observation O^{s2} , $S1$ and $S2$ denote the expert surgeon 1 and 2, respectively.

The performance was measured by $P(O|\lambda_s)$, the higher probability indicated the closer to the expert surgeon's performance. However, HMM tends to give higher likelihood to shorter observation sequences. In order to compare the observation sequences having difference observation frames, normalization is required. The simplest normalization is normalizing by the number of observation frames. However, it is not suitable for non-periodic observations [107]. Megali et al. [108] proposed to normalize by the mean

value of the log-likelihood L_m of the optimal observation sequence \hat{O}_i^s generated by the expert surgeon's model λ_s , the mean value of the log likelihood L_m was expressed as

$$L_m(O, \lambda_s) = \frac{1}{n} \sum_{i=1}^N \log P(\hat{O}_i^s | \lambda_s), \quad (2.25)$$

where N is the length of the observation sequence to be evaluated.

Hence, the evaluation metric was written as in [108]:

$$S_m(O, \lambda_s) = \frac{|\log P(O | \lambda_s) - L_m(O, \lambda_s)|}{\frac{1}{n} \sum_{i=1}^N |\log P(\hat{O}_i^s | \lambda_s) - L_m(O, \lambda_s)|}. \quad (2.26)$$

The denominator indicated the average distance of the optimal observation sequence from the mean log likelihood L_m .

Qiang et al. [107] proposed a relative HMM method in scoring the performance of the surgeon. Instead of training HMMs with explicit class labelled training sequences, the following was proposed by Qiang et al. [107],

$$\begin{aligned} \lambda : \max_{\lambda} \prod_{\mathbf{O}^i \in \mathcal{O}} P(\mathbf{O}^i | \theta), \\ s.t.: P(\mathbf{O}^i | \lambda) > P(\mathbf{O}^j | \lambda), \forall (i, j) \in \mathbb{E}, \end{aligned} \quad (2.27)$$

where $P(\mathbf{O}^i | \lambda)$ is the observation likelihood with model $P(O^i | \lambda)$; it is a score function for observation data \mathbf{O} given by model λ . The purpose of the function in the model is to maintain the ranking of the observation data pair $\mathbf{O}^i, \mathbf{O}^j$. \mathcal{O} is the set of training sequences. \mathbb{E} is the set of given pairs with prior ranking constraints.

$P(\mathbf{O}^i | \lambda)$ is bias towards the shorter observation sequences, Qiang et al. [107] used the ratio of likelihood with different model to decide the label of the data, the author further improved the model as

$$\begin{aligned} \lambda_1, \lambda_2 : & \max_{\lambda_{1,2}} [\sum_{i \in \Xi_1} \log P(\mathbf{O}^i, \hat{z}^i | \lambda_1) + \sum_{j \in \Xi_2} \log P(\mathbf{O}^j, \tilde{z}^j | \lambda_2)], \\ s.t. : & \log \frac{P(\mathbf{O}^i, \hat{z}^i | \lambda_1)}{P(\mathbf{O}^i, \tilde{z}^i | \lambda_2)} > \log \frac{P(\mathbf{O}^j, \hat{z}^j | \lambda_1)}{P(\mathbf{O}^j, \tilde{z}^j | \lambda_2)}, \forall (i, j) \in \mathbb{E}, \end{aligned} \quad (2.28)$$

where Ξ_1 is the set of data associated with model λ_1 , \hat{z}^i is the optimal path for sequence \mathbf{O}^i with model λ_1 , \tilde{z}^i is the optimal path with model λ_2 .

2.5.2.2 Liner Discriminant Analysis Method

LDA [106] has been applied to distinguish the performance of expert surgeons and novice surgeons according to the subjects' basic psychomotor skill expertise. LDA approaches the problem by assuming that the conditional $P(O|y=0)$ and $P(O|y=1)$ are both normally distributed with mean and covariance of the respective class y . In the evaluation application, LDA is trained by the features extracted from the experienced surgeon's execution. In Lin et al.'s [106] application, PCA was applied to extract the principal components to reduce the dimensionality. LDA uses the training data to learn the distribution of components and classify the subjects into each group: experts or novices.

2.6 Summary

In this chapter, a comprehensive review was conducted on medical simulation, robotic in medical filed, robot learning from demonstrations,

intention recognition and performance evaluation which will be applied to build the IRAS training system.

Although there is extensive research on robot or machine-mediated motor skill training, the robots were all pre-programmed or pre-recorded to perform a defined task, such as handwriting or a specific motion. The trajectories are obtained either directly from the experts or computer generated models. The robot does not model or learn the skills, and hence the robot cannot provide assistance as the teacher does. To our best knowledge, there is no robot-assisted surgical training system with the robot learning the skill through an autonomous learning process. Therefore, we would like to model the skills from the spatial trajectory perspective and equip the robot with the capability of recognizing the intention of the novice surgeon when he / she is performing a task. With the knowledge of the skills and awareness at the user's intention, the robot could be built with capability of providing the necessary assistance.

Statistical and neural networks approaches have been reviewed for Learning from Demonstration. They could enable the robot to possess knowledge of a surgery in terms of motion trajectory. HMM and GMM methods were reviewed in details. The current approaches use a left to right model or single-chain cyclic model with a predefined number of motion primitives to model a task, and arbitrary numbers of interconnected motion primitives were not considered [62]. The probabilistic approach requires the demonstration trajectories to be classified. In order to make the learning process require fewer user interventions, a classifier that can account for the characteristic of the motion data is required to identify the motion primitives.

Intention recognition is largely a problem of inference under conditions of uncertainties [92]. Researchers are trying to use the statistical method to model the mental process of the user while he / she is carrying out some specific tasks. HMM, DBN and probabilistic state machine are reviewed for intention recognition.

The objective of the evaluation methods which are reviewed in Section 2.5 is to distinguish the performance between the expert surgeons and novice surgeons, or the trained and untrained subjects, or to determine the proficiency level of the skills. The essential concept in this evaluation is classification. HMM method is a suitable method for such evaluation. However, the bias of likelihood towards shorter observation sequence in HMM method need to be taken care of. Researchers applied different normalization methods to minimize the bias. Another important component in the evaluation process is the observation feature. The observation feature decides the perspective of the evaluation method.

3 IMAGE-GUIDED AND ROBOT-ASSISTED SURGICAL TRAINING SYSTEM

This chapter presents our robot-assisted surgical training system known as the IRAS training system. In Section 3.1, the implementation and functions of the Image guided robot-assisted Surgical (IRAS) training system is explained. The current system is intended for laparoscopic skill training. It is capable of providing active guidance for surgical training. In Section 3.2, the design and development of the robotic surgical trainer, kinematics and control implementation are discussed. Subsequently, Section 3.3 discusses the friction mitigation for haptic rendering. Section 3.4 discusses the performance of the robotic surgical trainer through experimental validation. The friction mitigation method is also evaluated in this section. Section 3.5 summarizes the work in this chapter.

3.1 IRAS System

The IRAS system comprises of the robotic surgical trainer and the surgical simulation platform as shown in Figure 3-1. The system allows a user to conduct a virtual laparoscopic procedure by operating on a virtual patient through the robotic surgical trainer. The virtual laparoscopic procedure can be acquired and reproduced for training and analysis purposes.

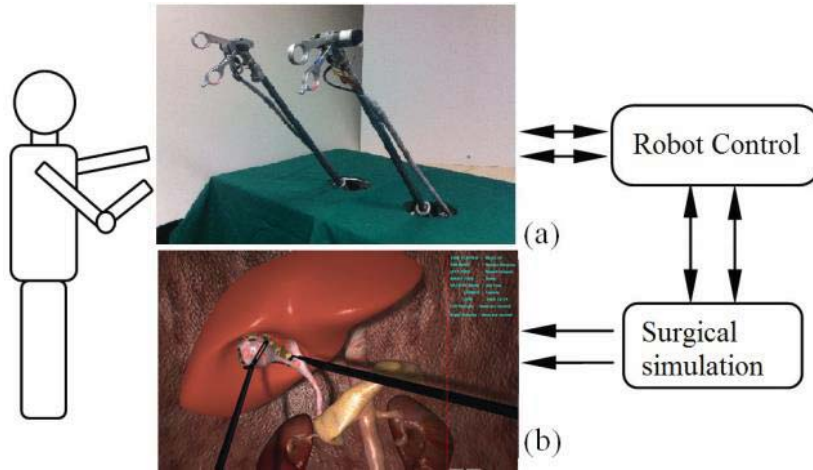


Figure 3-1 Overview of the IRAS surgical training system: (a) robotic surgical trainer and (b) virtual surgical simulation platform.

The robotic surgical trainer serves as a human-machine interface in both processes of acquiring surgical procedure and providing guidance to the users [114] in a training process. The robot was designed with two robotic laparoscopic instruments. Each of the instruments has 5 DOFs, namely pitch, yaw, roll, translation and grasping motion of the handle as indicated in Figure 3-3. It is capable of mimicking the motion kinematics of the laparoscopic instruments in real surgery. Users can operate with the robotic handles (Figure 3-1 (a)) and use them to perform a virtual surgery. The motion information of the robotic handles is sent to the surgical simulation platform to drive the virtual instruments and operate on the virtual patient. Motion trajectories of the robotic handles and virtual instruments, and status of the tool-organ interaction are recorded for the purpose of training and analysis. The robotic surgical trainer is one of the key components in the IRAS training system. The details are explained in Section 3.2.

The surgical simulation platform comprises of virtual patients, a tool library of laparoscopic instruments and a physics simulation engine. The tool library contains common instruments required for laparoscopic surgery, such as

forceps with different shapes and sizes, hook electrode, clip applicator and scissors. Tool-tissue interactions, organ deformation, tissue division, deployment of clips and other activities executed during surgery are simulated in the surgical simulation platform. In order to enhance the training performance of the system, the surgical simulation platform incorporates smoking, bleeding, perfusion and audio effects for the operations involving hook electrodes and scissors. Activation of bleeding and perfusion effects is triggered by remarkable events during the simulation, such as the collision between the objects, angle threshold of handle opening, contact pressure between the tool tip and organ surface. A basic assessment is provided after every surgical simulation; including time spent, average velocity of the tool tip, number of bleeding sites and perfusions that occurred.

A simulated surgical procedure can be reproduced for training and demonstration. Motion of the robotic handle and tool-tissue interaction can be replayed on the robotic surgical trainer and the surgical simulation platform simultaneously. The user can hold on to the moving robotic handles while watching the simulated surgical procedure to appreciate the manoeuvres conducted by the experienced surgeon. Motor skills training can be conducted through such a record and replay procedure.

The IRAS training system is designed for patient-specific laparoscopic surgery training and simulation in which a model of any patient can be generated based on CT data and configured for a virtual surgery [115]. This allows medical residents to be exposed to a variety of surgical cases and provides them with a preview of any variation in anatomy before they start the

real surgery. A framework has been established to generate patient-specific anatomical models for virtual surgical procedure [116].

3.2 Robotic Surgical Trainer for Laparoscopic Surgery

The robotic surgical trainer serves as a human-machine interface in the IRAS system. In order for the robotic surgical trainer to function as an effective training platform, it should have the following two capabilities:

1. The robotic surgical trainer shall allow the user to operate on a virtual patient by manipulating a robotic device. This capability allows the experienced surgeon to pass the surgical skills to the robot, and also allows the novice surgeon to practice on the robot; and
2. The robotic surgical trainer shall be able to execute a trajectory. This capability enables the robot to provide guidance to the user for training purposes.

The robotic surgical trainer was designed with the above two requirements. The design considerations, kinematics, control hardware and methods will be discussed in this section.

3.2.1 Design Considerations

Laparoscopic instruments are long and slender tools. Its applicator is driven by a lever mechanism through a handle. Generally, the mobility of a laparoscopic instrument constrained at the insertion point (trocar) includes five DOFs, namely roll, pitch, yaw, translation and grasping motion of the handle. (Trocar is a surgical instrument in laparoscopic surgery. It is placed on the abdominal wall of the patient. Surgical tools are inserted into the abdominal

cavity through the trocar.) Figure 3-2 illustrates the mobility of one laparoscopic instrument during the surgical process. The robotic surgical trainer which consists of two manipulators representing the surgical instruments is designed with the five DOFs to fulfill the required mobility. Details of the kinematics of the mechanism are presented in Section 3.2.2.

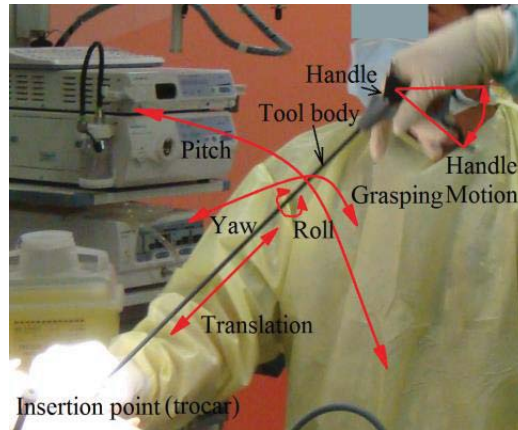


Figure 3-2 Motion of surgical instrument in laparoscopy procedure.

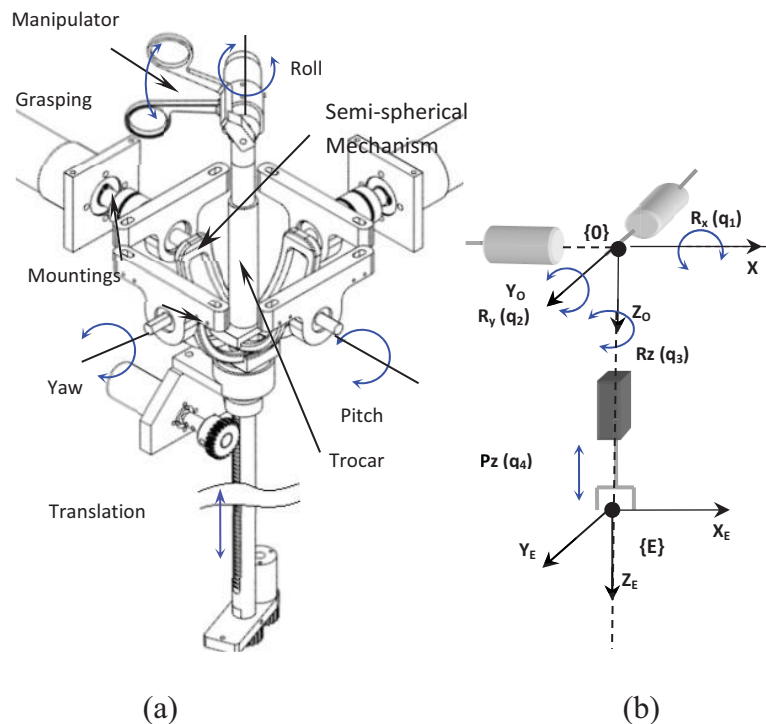


Figure 3-3 (a) Mechanical mobility of the robot. The travelling limit for pitch, yaw, roll, translation, handle grasping motion are 120°, 120°, 360°, 350mm and 60°, respectively. (b) Kinematic model of surgical instrument.

The mechanism, as shown in Figure 3-3 (a), is designed to mimic the kinematic motion of laparoscopic instruments. Spherical mechanism, rack and pinion system, and modified instrument handle, which are highlighted in Figure 3-4, have been used. The mobility of the mechanism is designed as an exact mimic of the kinematics of laparoscopic procedure. Since the task space of laparoscopic procedure can be readily expressed in spherical coordinates, an ideal mechanism design will be one with axes of control corresponding to the spherical coordinates. The orientation joints should coincident at a point for optimal geometric workspace efficiency [117]. Hence, the hybrid spherical mechanism as shown in Figure 3-4 is adopted. This hybrid mechanism possesses the advantages of both serial and parallel manipulator as explained in [117]. Similarly, this spherical mechanism is advantageous for both hardware and software performance. Unlike most general manipulators, the spherical mechanism is highly decoupled with each of the actuated axes of control corresponding to task-oriented space coordinates. This direct mapping of joint space to task-oriented space allows high frequency control and enables fast data updating from the actuation and sensory unit to the graphic and haptic rendering module without being burdened by space domain transformation.

In laparoscopic surgery, the instrument is manipulated at substantial moment arm about the insertion point (trocar). This requires high operational torque range for the pitch and yaw axis of control. Although closed-chain mechanism generally provides structural stability, the structural redundancy is workspace inefficient and collision prone with at least two manipulators simulating the laparoscopic instruments. The argument for parallel mechanism

to improve manipulator stiffness is therefore ineffective as it either increases structural link length index or reduces workspace [118]. This may also raise safety concern in user interaction due to the high manipulator stiffness associated with parallel linkage configuration. The proposed structure enables a more even and appropriate sizing of actuators for the range of operation configurations.

Apart from kinematic requirements, user centric design attributes like ergonomics and usability are considered. The modified handle is the only physical user interface in the user workspace. All other actuation and control mechanisms are concealed underneath the insertion point (trocar). This user centric configuration produces a more realistic operating environment during training.

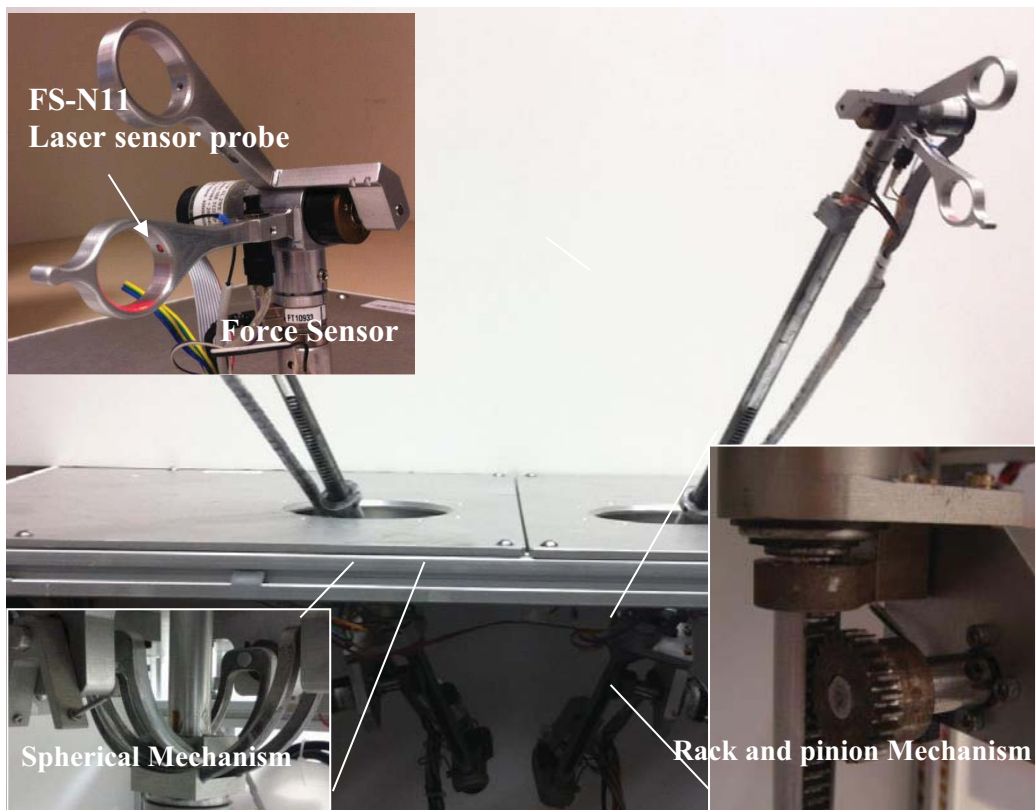


Figure 3-4 Details of the robotic surgical trainer.

The force range for haptic feedback on each axis is designed for general laparoscopic simulation. Existing literature has shown that the maximum pulling forces along the translation direction, grasping and cutting force are about 17 N, 16 N [110] and 14 N [119], respectively. The actuators are specified based on these guidelines to permit wide range of haptic feedback.

3.2.2 Kinematic Analysis

The designed robotic laparoscopic surgical instrument has 5 DOFs. However, the grasping motion of the handle does not contribute to the kinematic analysis. Hence, the grasping motion of the handle is not included in the kinematic analysis. In a laparoscopic procedure, the task space configuration can be defined with four DOFs through Euler angles, pitch, yaw, roll and translation $(\alpha, \beta, \gamma, \rho)$. Figure 3-3 (b) illustrates the frame assignment of the multibody system for kinematic analysis. The actuators are mounted such that their axes of control are aligned to the respective axis of transformation in our robot. Hence, the joint variables (q_1, q_2, q_3, q_4) correspond to the Euler angles and translation $(\alpha, \beta, \gamma, \rho)$.

Homogenous transformation matrix (3.1) expresses the forward kinematics of Frame E in Cartesian coordinates,

$${}^0\mathbf{T}_E = \begin{bmatrix} \hat{X}_x & \hat{Y}_x & \hat{Z}_x & E_x \\ \hat{X}_y & \hat{Y}_y & \hat{Z}_y & E_y \\ \hat{X}_z & \hat{Y}_z & \hat{Z}_z & E_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} s_1s_2s_3 + c_2c_3 & s_1s_2c_3 - c_2s_3 & c_1s_2 & q_4(c_1s_2) \\ c_1s_3 & c_1c_3 & -s_1 & -q_4s_1 \\ s_1c_2s_3 - s_2c_3 & s_1c_2c_3 + s_2s_3 & c_1c_2 & q_4(c_1c_2) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.1)$$

where $s_i, c_i, i=1,2,3$ denote $\sin(q_i)$ and $\cos(q_i)$, respectively.

The Jacobian matrix to map the joint space (q_1, q_2, q_3, q_4) into task space

is formulated as shown in Equation (3.2):

$$\mathbf{J}_E = \begin{bmatrix} -q_4(s_1s_2) & q_4(c_1c_2) & 0 & c_1s_2 \\ -q_4c_1 & 0 & 0 & -s_1 \\ -q_4(s_1c_2) & -q_4(c_1s_2) & 0 & c_1c_2 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \quad (3.2)$$

The force sensor is located right below the instrument handle, the distance between the handle and force sensor is ignored during kinematic modelling. With a given homogeneous matrix acquired from the sensory unit ${}^0\mathbf{T}_E$, the inverse kinematics is listed as follows:

$$\begin{aligned} q_1 &= \arctan 2(-E_y, E_z / \cos q_2), \\ q_2 &= \arctan 2(E_x, E_z), \\ q_3 &= \arctan 2(\hat{X}_y, \hat{Y}_y), \\ q_4 &= E_x / (\cos q_1 \cdot \sin q_2). \end{aligned} \quad (3.3)$$

3.2.3 Dynamics Analysis

The robot has only one prismatic link q_4 composed of the rotational joints, the dynamic model is simplified as shown in Figure 3-5.

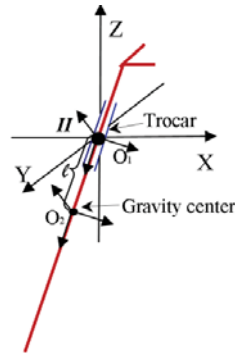


Figure 3-5 Dynamic model of the robot in inertia frame II . l is the distance from the gravity center of tool to the trocar. O_1 is the rotational link attached to the center of the semi-spherical mechanism (see Figure 3-3). O_2 is the translational link attached to the gravity center of the prismatic component.

For simplicity of discussion, the mass of handle hoop is ignored during modelling, therefore the inertia tensor of the laparoscopic instrument is a diagonal matrix and expressed as

$$\mathbf{I} = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix}. \quad (3.4)$$

The mass inertia matrix $\mathbf{D}(q_i)$ is expressed as

$$\mathbf{D}(q) = \begin{bmatrix} D_{11} & D_{12} & D_{13} & 0 \\ D_{21} & D_{22} & D_{23} & 0 \\ D_{31} & D_{32} & D_{33} & 0 \\ 0 & 0 & 0 & D_{44} \end{bmatrix}, \quad (3.5)$$

where $D_{11} = I_x c_2^2 + I_z s_2^2 + m q_4^2 c_2^2$, $D_{12} = D_{21} = -I_z s_1 s_2$,

$$D_{13} = D_{31} = I_z s_2, \quad D_{22} = I_x c_1^2 + I_z s_1^2 + m q_4^2 c_1^2,$$

$$D_{23} = D_{32} = -I_z s_1, \quad D_{33} = I_z,$$

$$D_{44} = m,$$

m is the mass of the laparoscopic instrument.

The Coriolis and centrifugal force vector \mathbf{u} is expressed as

$$u_1(q, \dot{q})\dot{q} = 2(I_z - I_x - ml^2)c_2 s_2 \dot{q}_1 \dot{q}_2 + 2mlc_2^2 \dot{q}_1 \dot{q}_4 + (I_x - I_z + ml^2)c_1 s_1 \dot{q}_2 \\ + I_z(c_1 \dot{q}_2 \dot{q}_3 + c_2 q_2 q_3 - s_1 c_2 \dot{q}_2^2),$$

$$u_2(q, \dot{q})\dot{q} = 2(I_z - I_x - ml^2)c_1 s_1 \dot{q}_1 \dot{q}_2 + 2mlc_1^2 \dot{q}_1 \dot{q}_4 + (I_x + ml^2)c_2 s_2 \dot{q}_1^2 \\ - I_z(c_1 s_2 \dot{q}_1^2 + c_1 \dot{q}_1 \dot{q}_3 + c_2 \dot{q}_1 \dot{q}_3 + c_2 s_2 \dot{q}_1^2),$$

$$u_3(q, \dot{q})\dot{q} = I_z(c_1 - c_2)\dot{q}_1 \dot{q}_2,$$

$$u_4(q, \dot{q})\dot{q} = -m(l\dot{q}_1^2 + c_1^2 \dot{q}_2^2). \quad (3.6)$$

The potential energy vector \mathbf{g} is expressed as

$$\begin{aligned}
g_1(q) &= -m l g s_1 c_2, \\
g_2(q) &= -m l g c_1 s_2, \\
g_3(q) &= 0, \\
g_4(q) &= m l g c_1 c_2,
\end{aligned} \tag{3.7}$$

3.2.4 Control Hardware

The robotic surgical trainer is programmed in Labview 2010 and controlled with National Instrument CompactRIO. The CompactRIO consists of Xilinx Virtex-5 LX110 reconfigurable I/O FPGA core and real-time embedded controller with 400 MHz processor, 128 MB DRAM memory. Each robotic arm is equipped with a six DOF high precision force sensing unit, ATI Nano17. The force sensor is calibrated at a force resolution of 0.0125 N and torque resolution of 0.0625 N.mm. A Keyence FS-N11 laser sensor is installed on each robotic handle to detect the presence of the user. Each of the 5 DOFs was driven by one actuator. The resolutions of the actuators are listed in Table 3-1. The robot is able to execute a high accuracy in motion trajectory with the high resolution from the actuators.

Table 3-1 Resolution of the actuators for each DOF

	Pitch	Yaw	Translation	Roll	Grasping
Resolution (Degree)	0.008	0.008	0.015	0.008	0.008

This hardware configuration allows high speed control loop execution, and ensures task determinism for managing communication flow crucial to haptic fidelity and human-machine interface applications in the IRAS system. The FPGA based real-time hardware platform is effective in the implementation of reliable controls including force feedback signal processing. The parallelism

nature of the FPGA operation mechanism facilitated fast and robust coordination amongst axes simplifying the issue of joint synchronization.

Control operation and computational task were mostly hard programmed in FPGA. This allows minimal delay in the compensation of the parasitic forces. Control of the manipulator is implemented with FPGA at a rate of 20 kHz to ensure determinism and maintain fidelity.

3.2.5 Control Methods

The robot is designed to work with two modalities, i.e. the *active mode* and the *passive mode*, to meet the two requirements stated in the beginning of this section. The passive mode is defined as the robot being operated by the user. The user takes the robotic surgical trainer as surgical platform for the virtual patient. In the passive mode, the robotic surgical trainer passes the position signal of the laparoscopic instruments to the surgical simulation platform. While the user performs a virtual surgery, the IRAS system records the procedure, includes trajectories of the robotic handles and virtual instruments, and statues of the tool-organ interaction. The active mode is defined as the robot executing a recorded or preplanned trajectory for complete guidance for surgical training. The user takes the robotic surgical trainer as a training provider.

Dynamics of the robot with non-rigid body effects, such as the parasitic forces, expressed in joint space as

$$\mathbf{D}(q)\ddot{q} + \mathbf{u}(q, \dot{q})\dot{q} + \mathbf{g}(q) + \mathbf{f}_{dis}(q, \dot{q}) + \mathbf{f}_{hap} = \tau, \quad (3.8)$$

where \mathbf{D} is the inertial tensor obtained from the kinetic energy of the robot expressed in Equation (3.5); \mathbf{u} is the coriolis and centrifugal force vector expressed by Equation (3.6); \mathbf{g} is the potential energy vector expressed by

(3.7); \mathbf{f}_{dis} is the non-rigid force, such as frictional force. \mathbf{f}_{hap} is the interactive force for haptic output. τ is the control law for control design. Haptic and frictional forces are discussed in Section 3.3. The two modalities are achieved through different control scheme.

Position control was implemented for the *active mode*. The controller for position control was designed using the computed torque method by setting the

$$\tau = \alpha u + \beta, \quad (3.9)$$

with $u = \ddot{q}$ and $\beta = \mathbf{u}(q, \dot{q})\dot{q} + \mathbf{g}(q) + \mathbf{f}_{dis}(q, \dot{q})$. α is the inertial tenor \mathbf{D} in this case. \mathbf{f}_{hap} is not applicable in pure trajectory tracking task.

Figure 3-6 shows the detailed diagram for the controller described in Equation (3.9). q_d is the desired trajectory parameter in joint space.

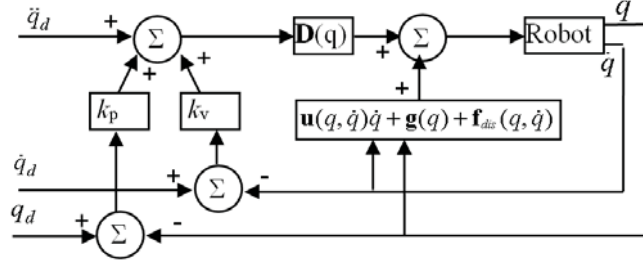


Figure 3-6 Implementation block diagram of the position control for active mode of the robot.

A PID position control is implemented to reproduce the master surgeon's trajectory. All actuators for joints (q_1, q_2, q_3, q_4) and handle grasping joint are commanded to move as per the desired trajectory and velocity which are acquired from the master surgeon's operation. A position control is appropriate as there is no need for variation in intensity of guidance. In the

active mode, the robot follows the trajectory planned or recorded by the expert surgeon.

Force control was implemented for the *passive mode*. In the *passive mode*, the robot is operated by the user, and it outputs haptic force. The dynamical equation for the robot with haptic output expressed in joint space is expressed as

$$mk_e^{-1}\ddot{\mathbf{f}}_e + \mathbf{f}_e + \mathbf{f}_{dis} = \mathbf{f}, \quad (3.10)$$

where \mathbf{f}_e is force acting with the environment, in our case it is the haptic force acting with the user, $\mathbf{f}_e = \mathbf{f}_{hap}$, $k_e = k_{hap}$ is the stiffness from the environment.

With the controller $\mathbf{f} = \alpha u + \beta$ and Equation (3.10), the parameter in the controller are expressed as

$$\begin{aligned} u &= \ddot{\mathbf{f}}_{hap} + k_{vf}\dot{e}_f + k_{pf}e_f, \\ \beta &= \mathbf{f}_{dis} + \mathbf{f}_{hap}, \\ \alpha &= mk_{hap}^{-1}, \end{aligned} \quad (3.11)$$

where $e_f = \mathbf{f}_{des} - \mathbf{f}_{hap}$ is the force error between the desired haptic output force \mathbf{f}_{des} and the interaction haptic force measured by the force sensor \mathbf{f}_{hap} .

With Equation (3.10) and (3.11), the system dynamical equation can be rewritten as

$$mk_e^{-1}\ddot{\mathbf{f}}_{hap} + \mathbf{f}_{hap} + \mathbf{f}_{dis} = mk_{hap}^{-1}(\ddot{\mathbf{f}}_{des} + k_{vf}\dot{e}_f + k_{pf}e_f) + \mathbf{f}_{des}. \quad (3.12)$$

Figure 3-7 shows the block diagram for Equation (3.12). However, in the real implementation, the haptic force variation is assumed noncritical, i.e. $\dot{\mathbf{f}}_{des} = \ddot{\mathbf{f}}_{des} = 0$. Hence, the block diagram shown in Figure 3-7 can be further simplified during implementation. The implemented control diagram with frictional force compensation is shown in Figure 3-8.

compensation has been studied thoroughly in the past decades [120]. Various friction models have been proposed and tested. However, the friction is difficult to model and compensate in reality due to the nature of wear and tear.

Frictional force between moving parts can be categorized as two basic categories: rolling friction and sliding friction,

$$\text{Rolling friction } \mathbf{f}_{roll} = \mathbf{c}_{rr} \mathbf{f}_N, \quad (3.14)$$

$$\text{Sliding friction } \mathbf{f}_{slid} = \boldsymbol{\mu}_s \mathbf{f}_N, \quad (3.15)$$

where \mathbf{c}_{rr} is the rolling resistance coefficient, and $\boldsymbol{\mu}_s$ is the sliding frictional coefficient. \mathbf{f}_N is the normal force acting on the contact force. In our case, \mathbf{f}_N varies as the haptic output varies.

The rolling resistance coefficient \mathbf{c}_{rr} depends on material elasticity, in other words, the deformation of the moving part. The sliding frictional coefficient $\boldsymbol{\mu}_s$ depends on material pair and surface condition. $\boldsymbol{\mu}_s$ is usually much larger than \mathbf{c}_{rr} . Special material pair produces very low frictional coefficient. Therefore, it is always preferred to have rolling motion for all possible moving parts when friction is a concern.

The relationship between the stable state friction and velocity is expressed as in [121]:

$$\mathbf{f}_{ss} = (\mathbf{f}_c + (\mathbf{f}_s - \mathbf{f}_c) e^{-(\mathbf{v}/\mathbf{v}_s)^2}) \text{sgn}(\mathbf{v}), \quad (3.16)$$

where \mathbf{f}_{ss} is the steady state friction. \mathbf{f}_c is the Coulomb frictional force, \mathbf{f}_s is the stribek force, \mathbf{v}_s is the relative velocity at stribek, and \mathbf{v} is the relative velocity of two moving components.

In the robot system, the haptic output can cause the change of contact force of two moving parts, and hence the frictional force. Therefore, \mathbf{f}_c and \mathbf{f}_s are dependent on the magnitude of haptic output and the relative velocity of the two moving components in our application. They can be written as a function of desired haptic output \mathbf{f}_{des} . A second order polynomial equation is applied to represent the stricbeck force in the system \mathbf{f}_s . Hence, Equation (3.16) is expanded as

$$\mathbf{f}_{ss} = [\mu_e \mathbf{f}_{des} + [(a_2 \mathbf{f}_{des}^2 + a_1 \mathbf{f}_{des} + a_0) - \mu_e \mathbf{f}_{des}] e^{-(v/vs)^2}] \text{sgn}(v), \quad (3.17)$$

where μ_e is the equivalent frictional coefficient for the system; In a complex system which consists of both sliding friction and rolling friction, it is hard to be simply determined by the material pair. Furthermore, due to imperfectness of manufacturing, wear and tear, the frictional coefficient μ_e needs to be determined from experiments in real application. $a_2 \mathbf{f}_{des}^2 + a_1 \mathbf{f}_{des} + a_0$ is the polynomial introduced to represent the stricbeck frictional force \mathbf{f}_s . They will be determined in Section 3.4.2 by fitting Equation (3.17) with the measured frictional force.

Although our design has minimized the effect due to friction, friction compensation is still required for high haptic output that will lead to high friction between the moving parts. The resultant frictional force in our robot is a combination of sliding friction from the bushing and rolling friction from the rollers. Hence, stricbeck phenomena would affect the performance of haptic output, especially when the output force is large and moving velocity is low. In this study, a motion-based friction cancellation method [121] was applied to

compensate the frictional forces in the system. The control diagram of such compensation method and force control is as shown in Figure 3-8. Experiments were conducted to measure residual frictional force while robot was commanded to output haptic force with different magnitude. Experimental details are explained in Section 3.4.2.

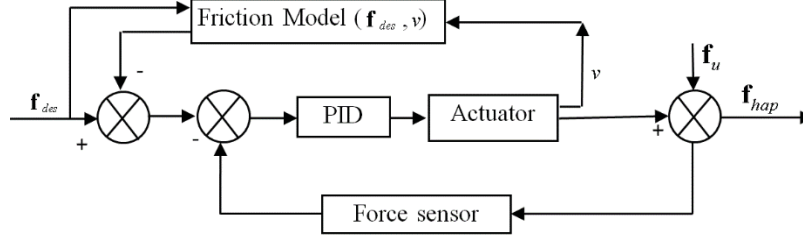


Figure 3-8 Control diagram for friction compensation and haptic output. \mathbf{f}_{des} is the desired haptic output reference force, \mathbf{f}_{hap} is the haptic output force, \mathbf{f}_u is the user's interaction force.

3.4 Experiments

3.4.1 Robotic Performance Analysis

The kinematics and dynamics profiles were acquired and analysed for a given path execution in a specified operational workspace. To evaluate the efficacy of the control mechanism, a recorded trajectory was executed by the robot under the condition with and without user interaction. Kinematic trajectories were acquired through the encoder with joint control scheme at a frequency of 100 samples / second and subsequently transformed to 3D Cartesian coordinates for analysis. The force profile was acquired by the 6-DOF force sensor through FPGA based DAQ module.

The robot was tested when it was working under complete guidance. When there was no interaction with the user, the maximum positional errors of

execution on the left and right manipulators were 2.12 mm and 1.55 mm, respectively. The robot was also tested by guiding a user to perform the recorded path. In order to minimize the effect of visual guidance, the trajectory was neither displayed nor known to the user prior to the test. Figure 3-9 depicts the 3D trajectory and the force profile in Cartesian coordinates when the robot was interacting with a user. The mechanical components and motion control mechanism are capable of precise and accurate execution. The maximum errors of execution on both right and left manipulators are 1.87 mm and 2 mm, respectively. The maximum errors of each joint in the left and right manipulators are tabulated in Table 3-1. This trajectory spanned an approximated $(0.2 \times 0.2 \times 0.2) \text{ m}^3$ workspace and was subjected to a maximum interaction force of 3.6 N and torque of 73.7 Nmm.

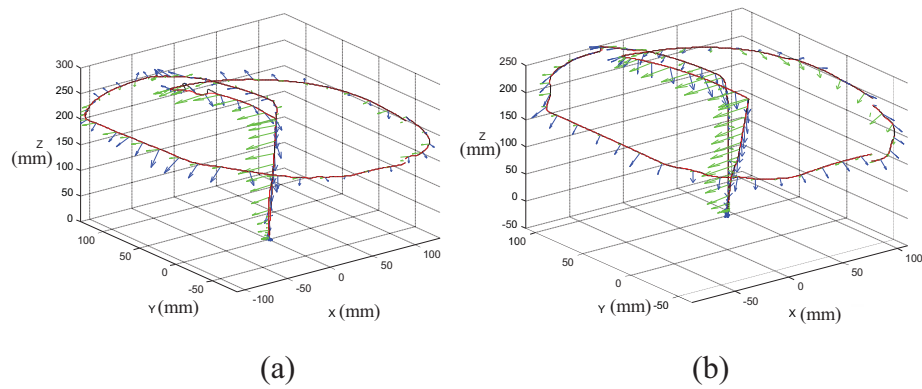


Figure 3-9 (a) Execution and force on handle of left manipulator, (b) execution and force on handle of right manipulator. Red line is the recorded trajectory, black line is the execution results, blue arrow indicates the force vector on handle, and green arrow indicates the moment vector on handles.

Table 3-2 Maximum positional error of each joint.

Joint	Manipulator	
	Left	Right
Pitch (q_1)	0.230°	0.425°
Yaw (q_2)	0.241°	0.313°
Roll (q_3)	1.66°	1.67°
Translation (q_4)	0.524 mm	1.35 mm

3.4.2 Experiment of Friction Mitigation for Haptic Rendering

Experiments were conducted to measure the frictional force on the pitch and yaw axes of the robot with and without friction compensation. The robot was set to output a series of haptic force exerted on a user. The haptic output was set from 1N to 7N with 1N increment for each experiment. The user pushed the robotic handle (as shown in Figure 3-4) to move against the direction of haptic output, and move the sliding block from one end of the arc to the other end as shown in Figure 3-10. The force was measured while the robotic handle was moving. Frictional force was obtained by subtracting the desired haptic output. This procedure was repeated for 50 times at each haptic output level. The velocity span was varied from 0 to 0.125 m/s. The maximum velocity in the experiment was relatively low. Therefore, viscous friction was not taken into consideration.

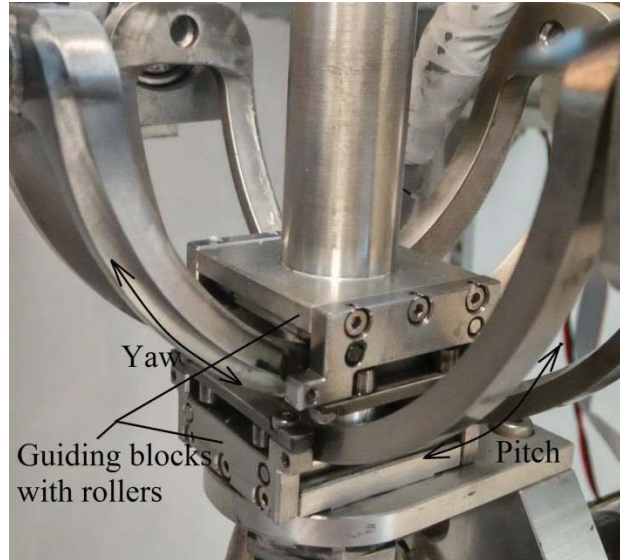


Figure 3-10 Relative velocity at the contacting area of each axis. The velocity measured was the relative linear velocity at the contacting area.

Figure 3-11 (a) and (b) show the measured mean frictional force along pitch and yaw axis, respectively. The red vertical bars show the standard derivation of the frictional force at the respective velocity and haptic output level. The frictional force ramped up when the object started to move, and reached its maximum at a velocity of 0.02m/s approximately. This is the stribek velocity described in Equation (3.16). The frictional force started to drop after the stribek velocity. It is noticed that the frictional force is higher when the haptic output is increased. It is due to the normal contact force F_N acting on the friction surface is increased as the haptic output is increasing. Surface fitting was performed to fit Equation (3.17) with the experimental data using a curve fitting toolbox in Matlab. Nonlinear least squares method was applied in the curve fitting process. Figure 3-12 (a), (b) and Table 3-2 show the fitting results.

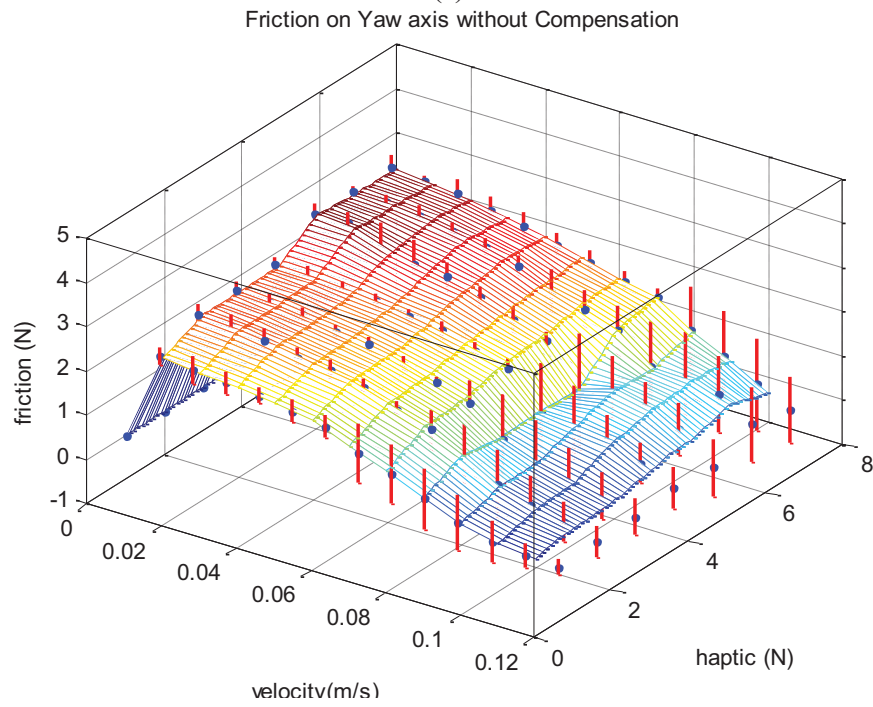
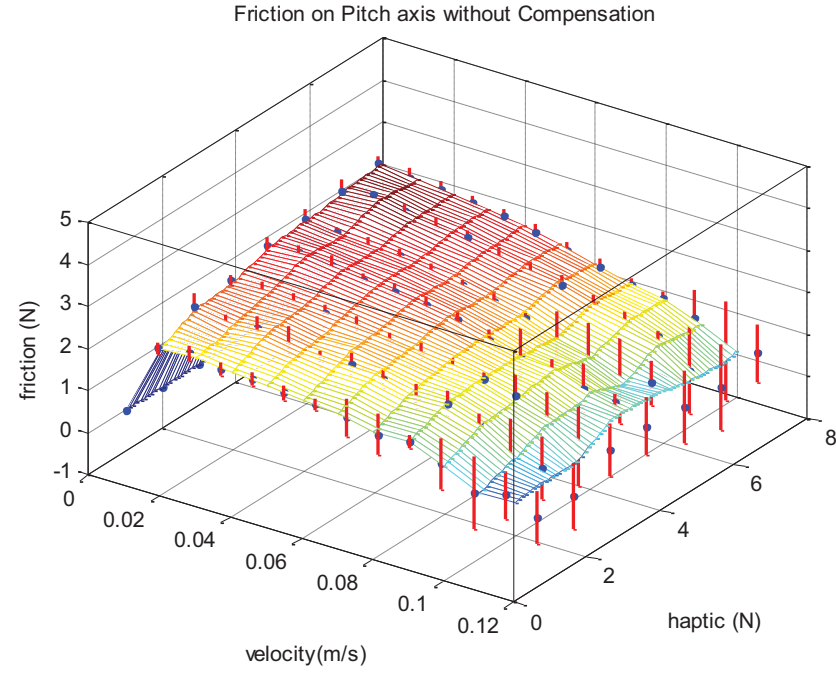
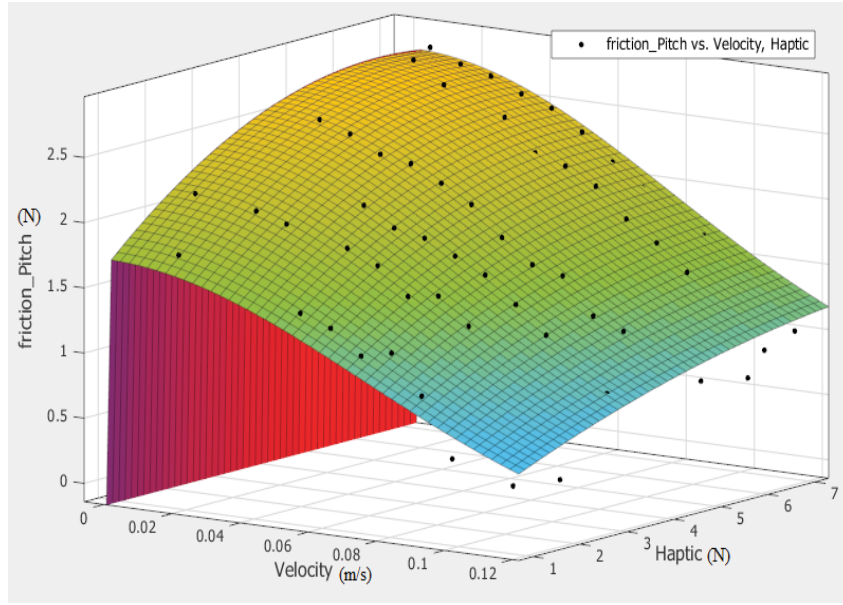
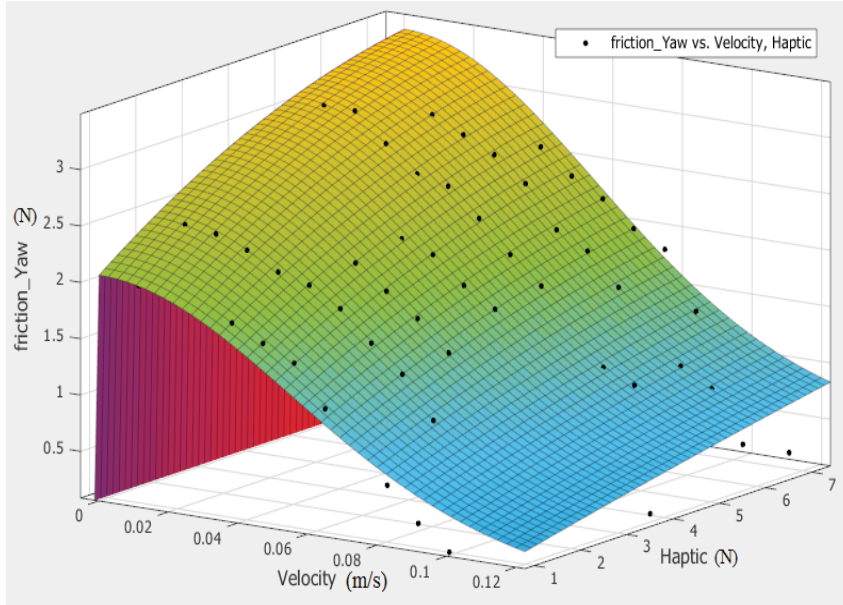


Figure 3-11 Mean frictional force from current design with haptic output from 1N to 7N. The frictional force is larger when the components are just to move, and it is reduced significantly and tends to stabilize when the components moving at higher velocity. The frictional forces are generally higher when the robot outputs a higher haptic force. Vertical bars are the standard deviations at the specific velocity and haptic output. (a) Frictional force for pitch axis. (b) Frictional force for yaw axis.



(a)



(b)

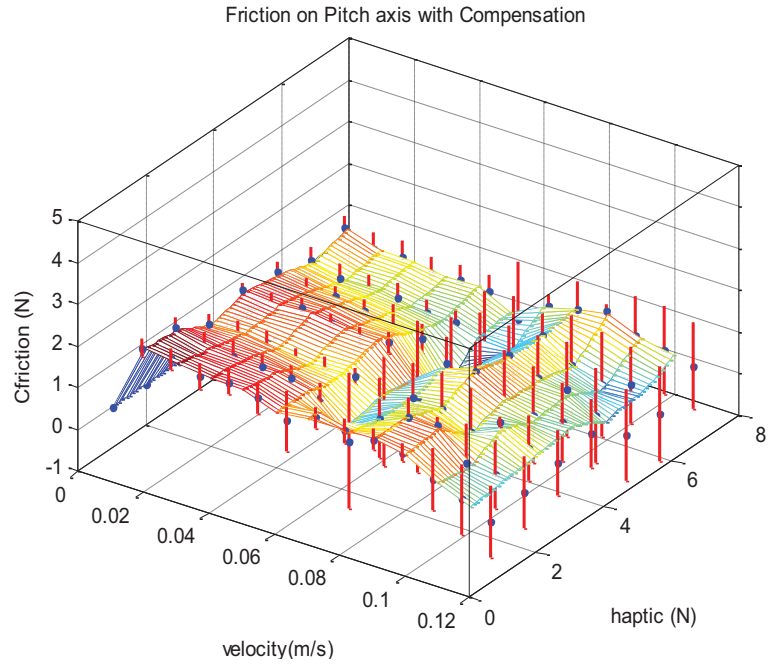
Figure 3-12 Surface fitting result with Equation (3.17). Experimental results shown in Figure 3-11 were fitted with Equation (3.17) using Matlab curve fitting toolbox. Black dots are the down sampled experimental measurements. The meshed surface is the fitting results. (a) Frictional force fitting for pitch axis with $\mu_e=0.08$, $a_2=-0.032$, $a_1=0.403$, $a_0=1.476$. (b) Frictional force fitting for yaw axis with $\mu_e=0.086$, $a_2=-0.019$, $a_1=0.351$, $a_0=1.82$.

Table 3-3. Frictional force fitting results with Equation (3.17).

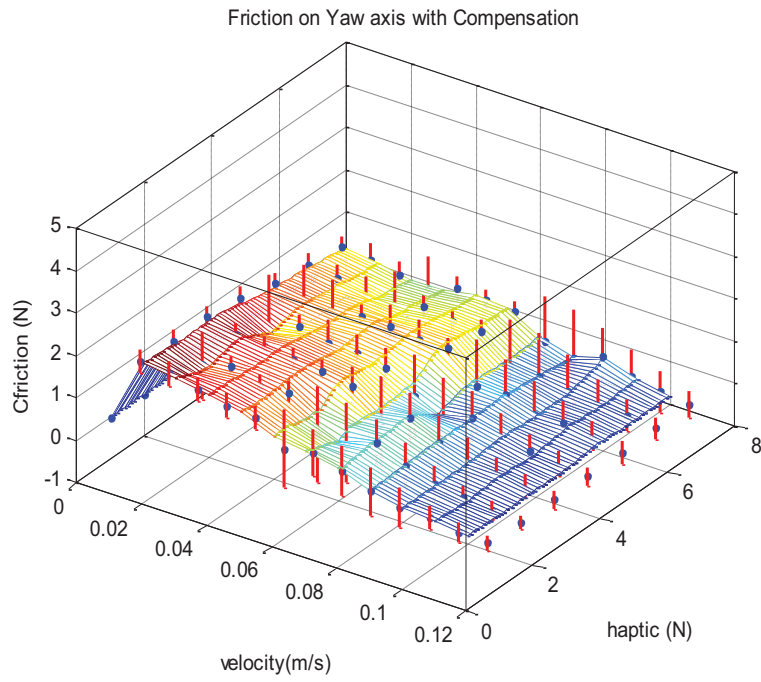
	R^2	Adjusted R^2	RMSE
Pitch	96.75%	96.65%	0.14
Yaw	93.92%	93.72%	0.24

Motion-based friction cancellation method, as shown in Figure 3-8, was implemented with the fitting results. Same experiments as described in the beginning of this section were conducted to illustrate the residual friction after compensation. Figure 3-13 showed the mean measured frictional force collected from experiments. Comparing with the frictional force shown in Figure 3-11 which has no compensation, the frictional force and the stiction phenomena have been mitigated significantly. The total volume covered under the surface were reduced by 49.46% and 62.08% for pitch and yaw axis, respectively. The maximum residual force is about 1N.

Although the frictional force has been mitigated significantly, some residual frictional forces are not removed due to the limitation of motion-based cancellation method. This is because the motion-based cancellation method needs a velocity input before it can estimate the friction for compensation, but the frictional force is already there before velocity is detected. The compensation is therefore always later than the actual frictional forces. Comprehensive system model and friction model are required to further improve the performance of this motion-based cancellation method. Advanced friction compensation methods need to be explored.



(a)



(b)

Figure 3-13 Mean residual frictional force measured with compensation. Vertical axis is the measured frictional force after compensation. Vertical bars are the standard deviations at the specific velocity and haptic output. (a) Frictional force for pitch axis. (b) Frictional force for yaw axis.

3.5 Summary

A new robotic surgical trainer for hand-by-hand guidance in laparoscopic surgical training is introduced in this chapter. The robot is designed with appropriate specifications and implemented as an interactive platform that can adequately meet the training needs of laparoscopic surgeries. The robot can be passively operated under the force control method, and it can also actively move itself by the position control method.

The robot is developed with the provision of complete guidance method capable of guiding the novice surgeon according to a recorded trajectory. This complete guidance method could provide the novice surgeon with deep appreciation of how an experienced surgeon deals with specified surgical scenarios. The robot is also designed with haptic features to render the force feedback for tool-tissue interaction. Frictional force of the system is measured at different levels of haptic output, and it has been mitigated significantly using a motion-based cancellation method.

4 MOTION MODELLING, LEARNING AND GUIDANCE

Robotic technologies have been widely applied in surgery, playing a significant role in robot-assisted surgery, teleoperation [2, 3] and robotic surgical training [4]. However, it is a challenging task for the robot to learn the complex manipulation of surgical instruments during surgery. To equip our robotic surgical trainer presented in the previous chapter with knowledge of surgical skills, the robot needs to have the capability of observing a surgical procedure, extracting the skill set, representing and then repeating the execution of the skills. In this chapter, the methods for representing and reconstructing the surgical skills are discussed.

A key technical contribution is the proposed motion learning method that uses adaptive mean shift method to identify the motion primitives. Section 4.1 introduces the proposed method for surgical motion trajectory learning and illustrates the various techniques in each module of the proposed method. Section 4.2 describes the application of this method with the experimental results of a tissue division task and a clip deployment task in the IRAS training system. In Section 4.3, the generic motion model which is trained using primitives determined by adaptive mean shift method is compared with that of K-means method and fixed bandwidth mean shift method. Finally, our motion modelling and learning approach is summarized in Section 4.4.

4.1 Methods

Figure 4-1 describes the proposed method for the clustering, modelling and reconstruction of the motion trajectory in the robotic learning of laparoscopic surgery.

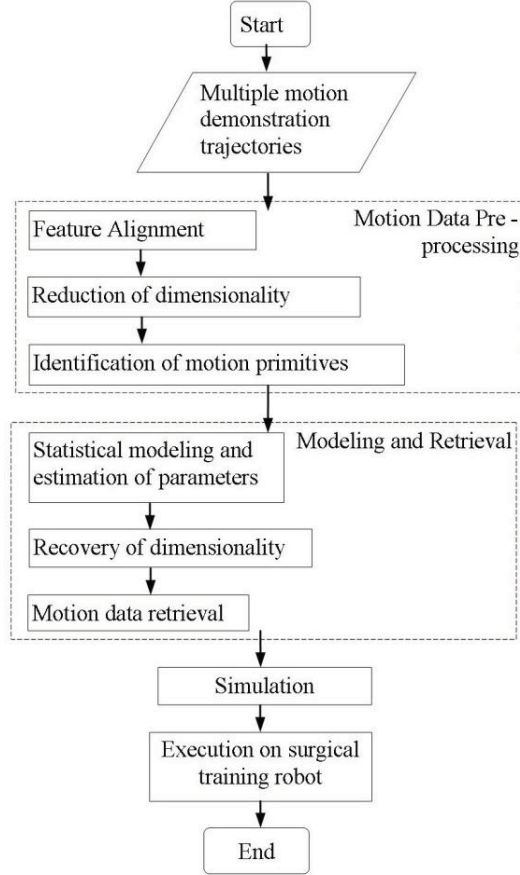


Figure 4-1 Data processing procedure to model and reconstruct the motion trajectories.

Suppose that the motion trajectory of the laparoscopic instrument is expressed as

$$\mathbf{X} = \{X_{t,i}, \mathbf{X}_{s,i}\}, i = 1 \cdots N, \quad (4.1)$$

where X_t and \mathbf{X}_s are the time and spatial components of the trajectory, respectively. N is the number of the observations. In order to eliminate the effect of the non-homogeneity of motion speed among the trials of the same

task, Dynamic Time Warping (DTW) is performed to align the trajectories according to its features. The Principal Component Analysis (PCA) is applied to reduce the dimensionality of the high dimensional data and to preserve their features. The aligned motion data sets are therefore transformed into latent space by the PCA. The motion data sets in latent space are then clustered by the adaptive mean shift method with optimal bandwidth to identify the motion primitives. The number of motion primitives is defined as the number of mixture components for statistical modelling. Gaussian Mixture Model (GMM) is trained with the clustered motion data to estimate its parameters. When the GMM model is trained, the estimated parameters in latent space are then projected back to their original space. With the motion data represented by GMM, Gaussian Mixture Regression (GMR) can be applied to retrieve smooth trajectory in original space with given temporal information.

4.1.1 Data Processing

Motion speed in execution of a given task varies from one trial to another. Therefore, the features in the motion trajectories do not appear in the same region across the trials. Hence, DTW is required to align the features from different trails in the same time span. The DTW measures the similarity between two trajectories which may vary in temporal information. It eliminates the constraint of distortions in time, between separate trajectories, which reduce the capability of the statistical models. To avoid misalignment during DTW, the trajectory data of each trial is divided into several subtasks with landmarks, such as approaching tissue, holding tissue, division of tissue. Each subtask is temporally aligned by the DTW. The trajectory candidate with

the longest time span is chosen as the reference trajectory during the DTW. The results of the DTW of each subtask are joined together accordingly, and expressed as $\mathbf{T} = \{T_{t,i}, \mathbf{T}_{s,i}\}$.

PCA is required to reduce the dimensionality for high dimensional data, reduce noise, and identify the principal axis of the temporal aligned trajectory data. With PCA, $\{\mathbf{T}_{s,i}\}$ is expressed in latent space. The spatial component in the latent space is written as

$$\{\mathbf{x}_{s,i}\} = \mathbf{A}_p' \cdot \{\mathbf{T}_{s,i}\}, \quad i = 1 \cdots N, \quad (4.2)$$

where $\mathbf{A}_p = \{\nu_{1,d}, \nu_{2,d}, \cdots, \nu_{i,d}\}$ is a transformation matrix, and ν_i is the eigenvectors of the covariance matrix of the centered motion data set $\{\mathbf{T}_{s,i}\}$ [61], and subscript d is the minimum number of the dimensionality required in the latent space. Hence, the motion trajectory data after PCA can be expressed as

$$\mathbf{x} = \{x_{t,i}, \mathbf{x}_{s,i}\}, \quad i = 1 \cdots N, \quad (4.3)$$

where $\mathbf{x}_{s,i}$ is the spatial component expressed in the latent space.

4.1.2 Adaptive Mean Shift Clustering of Motion Trajectory

Mixture model is a mixture distribution that represents the probability distribution of the observations in the overall population. The number of mixture components K_p and the number of observations are two basic parameters for any mixture model. In this study, the number of motion primitives in a task is the number of mixture components used in modelling the task. Identification of motion primitives is required for application of

mixture model in modelling motion trajectories. However, the number of motion primitives is not known for a demonstration of real tasks. The adaptive mean shift method can be applied to cluster the motion trajectories and to identify the number of components based on the bandwidth of the data set.

The mean shift method first defines a window around each data point and computes the mean of the data points, after which the centre of the window is shifted to the mean according to the mean shift vector and the algorithm is repeated until the mean shift vector is less than a specified threshold value. The data points in the feature space are considered as a probability density function. Kernel function is applied to estimate the density. The kernel density estimation is a non-parametric way to estimate the density function of a random variable. The kernel $k(\mathbf{x})$ is a positive definite bounded function satisfying $\int K(\mathbf{x})d\mathbf{x}=1$ and $\int \mathbf{x}K(\mathbf{x})d\mathbf{x}=0$ [122]. Given a kernel $K(\mathbf{x})=k(\|\mathbf{x}\|^2)$ with bandwidth parameter h , the kernel density estimator for a given set of D -dimensional data is expressed as

$$f(x) = \frac{1}{Nh^d} \sum_{i=1}^N K\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right). \quad (4.4)$$

where d is the dimensionality of the data.

There are several variants of exact kernel function [122]. Research [123] had shown that the profile of the kernel is not crucial to the kernel density estimation. The quality of the kernel estimation depends on the value of the bandwidth h instead of the profile of the kernel. Although the kernel density estimation has been commonly applied in data analysis, the determination of the optimal choice of the bandwidth for the kernel is still an active research topic [123-125].

The adaptive bandwidth introduced by Comaniciu et al. [126] was applied in this study. The adaptive bandwidth, a non-random sequence of positive numbers, is expressed as

$$h(\mathbf{x}_i) = h_o \left[\frac{\psi}{f(\mathbf{x}_i)} \right]^{\frac{1}{2}}, i = 1 \dots N, \quad (4.5)$$

where ψ is the proportionality constant and defined as $\log \psi = N^{-1} \sum_{i=1}^N \log f(\mathbf{x}_i)$, and h_o is the initial bandwidth. The plug-in-rule methods [127] were applied to determine an appropriate initial bandwidth in this study.

With Equations (4.4) and (4.5), the density estimation function for the adaptive bandwidth is written as

$$f(\mathbf{x}) = \frac{1}{N h(\mathbf{x}_i)^d} \sum_{i=1}^N K\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h(\mathbf{x}_i)} \right\|^2\right). \quad (4.6)$$

Hence, the mean shift vector is expressed as

$$\mathbf{m}_v(\mathbf{x}) = \frac{\sum_{i=1}^N \frac{\mathbf{x}_i}{h^{d+2}} g\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h(\mathbf{x}_i)} \right\|^2\right)}{\sum_{i=1}^N \frac{1}{h^{d+2}} g\left(\left\| \frac{\mathbf{x} - \mathbf{x}_i}{h(\mathbf{x}_i)} \right\|^2\right)} - \mathbf{x}. \quad (4.7)$$

where $g(\mathbf{x}) = -K'(\mathbf{x})$. The details of the derivation of the Equation (4.7) is available in [126].

4.1.3 Statistical Modelling and Parameter Estimation

4.1.3.1 Gaussian Mixture Model

Gaussian Mixture Model is a linear superposition of K_p Gaussian components, defined by probability density function

$$P(\mathbf{x}_i) = \sum_{k_p=1}^{K_p} P(k_p) P(\mathbf{x}_i | k_p), \quad i = 1 \dots N, \quad (4.8)$$

where $P(k_p) = \pi_{k_p}$ is the prior, and

$$P(\mathbf{x}_i | k_p) = \mathcal{N}(\mathbf{x}_i; \mathbf{\mu}_{k_p}, \mathbf{\Sigma}_{k_p}) = \frac{1}{\sqrt{(2\pi)^d |\mathbf{\Sigma}_{k_p}|}} e^{-\frac{1}{2}[(\mathbf{x}_i - \mathbf{\mu}_{k_p})^T \mathbf{\Sigma}_{k_p}^{-1} (\mathbf{x}_i - \mathbf{\mu}_{k_p})]} \quad \text{is the conditional}$$

probability density functions for component k_p , and $P(\mathbf{x}_i)$ is a probability that the data point \mathbf{x}_i is constructed by the model.

The parameters of the GMM are expressed as: $\{\pi_{k_p}, \mathbf{\mu}_{k_p}, \mathbf{\Sigma}_{k_p}\}_{k_p=1}^{K_p}$, where π_{k_p} is the prior probability, μ_{k_p} is the mean vector, and Σ_{k_p} is the covariance matrix.

The cumulated posterior probability of the GMM is expressed as

$$E_{k_p} = \sum_{i=1}^N P(k_p | \mathbf{x}_i). \quad \text{The number of the components } K_p \text{ is obtained by the}$$

adaptive mean shift clustering method described above. The trajectory data \mathbf{x}_i

contains the temporal and the spatial information, as shown in Equation (4.2),

and hence the mean vector is expressed as $\mathbf{\mu}_{k_p} = \{\mu_{t,k_p}, \mathbf{\mu}_{s,k_p}\}$, and the

$$\text{covariance matrix can be expressed as } \mathbf{\Sigma}_{k_p} = \begin{pmatrix} \Sigma_{tt,k_p} & \Sigma_{ts,k_p} \\ \Sigma_{st,k_p} & \Sigma_{ss,k_p} \end{pmatrix}.$$

The GMM parameters $\{\pi_{k_p}, \mathbf{\mu}_{k_p}, \mathbf{\Sigma}_{k_p}\}$ are estimated by Expectation Maximization algorithm [72] with the demonstration trajectory data in Equation (4.3). As the estimated parameters are for the data in the latent space and expressed as $\{\pi_{k_p}^{\prime\prime}, \mathbf{\mu}_{k_p}^{\prime\prime}, \mathbf{\Sigma}_{k_p}^{\prime\prime}\}$, they are projected back into the original space by

$$\begin{aligned}
\boldsymbol{\mu}_{k_p} &= \mathbf{A}_p \cdot \boldsymbol{\mu}_{k_p}'' \\
\boldsymbol{\Sigma}_{ss,k_p} &= \mathbf{B} \cdot \boldsymbol{\Sigma}_{ss,k_p}'' \cdot \mathbf{B}', \quad k_p = 1 \cdots K_p, \\
\pi_{k_p} &= \pi_{k_p}''
\end{aligned} \tag{4.9}$$

where π_{k_p}'' , $\boldsymbol{\mu}_{k_p}''$ and $\boldsymbol{\Sigma}_{ss,k_p}''$ are the prior probability, mean vector, covariance matrix of motion data set in the latent space respectively. The covariance matrix

in latent space is the form of $\boldsymbol{\Sigma}_{k_p}'' = \begin{pmatrix} \boldsymbol{\Sigma}_{tt,k_p}'' & \boldsymbol{\Sigma}_{ts,k_p}'' \\ \boldsymbol{\Sigma}_{st,k_p}'' & \boldsymbol{\Sigma}_{ss,k_p}'' \end{pmatrix}$. In order to recover the $\boldsymbol{\Sigma}_{ss,k_p}$ which

is covariance matrix of the spatial information, the transformation matrix is written in

the form $\mathbf{B} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_p \end{bmatrix}$, \mathbf{A}_p is the transformation matrix described in Equation (4.2).

4.1.3.2 Gaussian Mixture Regression

GMR is applied to reconstruct a trajectory represented by the GMM. The regression method estimates the conditional expectation of \mathbf{X}_s with given X_t , hence the entire trajectory can be reconstructed with its characteristics encoded by the GMM. For the k_p^{th} component at given time X_t , the expected distribution of \mathbf{X}_{s,k_p} is

$$P(\mathbf{X}_{s,k_p} | X_{t,k_p}) = N(\mathbf{X}_{s,k_p}; \mathbf{X}_{s,k_p}, \boldsymbol{\Sigma}_{ss,k_p}), \tag{4.10}$$

where \mathbf{X}_{s,k_p} and $\boldsymbol{\Sigma}_{ss,k_p}$ are the conditional expected value and expected covariance of the mixture component k_p , respectively. They are expressed as

$$\begin{aligned}
\mathbf{X}_{s,k_p} &= \boldsymbol{\mu}_{s,k_p} + \boldsymbol{\Sigma}_{st,k_p} (\boldsymbol{\Sigma}_{tt,k_p})^{-1} (X_t - \mu_{t,k_p}), \\
\boldsymbol{\Sigma}_{s,k_p} &= \boldsymbol{\Sigma}_{s,k_p} + \boldsymbol{\Sigma}_{st,k_p} (\boldsymbol{\Sigma}_{tt,k_p})^{-1} \boldsymbol{\Sigma}_{ts,k_p},
\end{aligned} \tag{4.11}$$

\mathbf{X}_{s,k_p} and Σ_{ss,k_p} are combined based on the probability that the component k_p for the given time X_t , which is expressed as

$$P(\mathbf{X}_s | X_t) = \sum_{k_p=1}^{K_p} \beta_{k_p} N(\mathbf{X}_{s,k_p}; \mathbf{X}_{s,k_p}, \Sigma_{ss,k_p}), \quad (4.12)$$

$$\text{where } \beta_{k_p} = \frac{P(k_p)P(X_t | k_p)}{\sum_{kp=1}^{K_p} P(i)P(X_t | i)} = \frac{\pi_{k_p} N(X_t; \mu_{t,k_p}, \Sigma_{tt,k_p})}{\sum_{kp=1}^{K_p} \pi_i N(X_t; \mu_{t,i}, \Sigma_{tt,i})}.$$

An estimation of the conditional expectation of \mathbf{X}_s at the given time X_t for the k_p^{th} component in the mixture model is

$$\mathbf{X}_s = \sum_{k_p=1}^{K_p} \beta_{k_p} \mathbf{X}_{s,k_p}, \quad \Sigma_{ss} = \sum_{k_p=1}^{K_p} \beta_{k_p}^2 \Sigma_{ss,k_p}, \quad (4.13)$$

The generalized form of the motion trajectory in its original space can be expressed as $\mathbf{X} = \{X_t, \mathbf{X}_s\}$.

4.2 Experiments and Results

Experiments were conducted to evaluate the proposed method. Three subjects (30 ± 3 years old) have participated in the experiments. Subject 1 performed a tissue division task, while Subject 2 and Subject 3 performed a clip deployment task. Tissue division and clip deployment are common tasks in surgical procedure; they are commonly found in laparoscopic cholecystectomy, sectionectomy of liver and colostomy etc. Based on our experience, more than 20 repeats of a demonstration will be sufficient for the purposes of modelling and analyses. Hence, 22 trajectories were collected from Subject 1 and Subject 2 each, and 24 trajectories from Subject 3. The motion trajectories collected from Subject 1's demonstration were used to show the feasibility of the method described in Section 4.1 with details. In the

following section, the experimental method and the modelling results are presented.

4.2.1 Experiments and Data Acquisition

Experiments were conducted on the IRAS system described in Section 3.1. The training system was built with a laparoscopic cholecystectomy procedure. A tissue division procedure and a clip deployment procedure within the cholecystectomy surgery were modelled in the experiment. In the tissue division procedure, as shown in Figure 4-2, a left hand laparoscopic grasper was used to stretch and hold a cystic duct, while the right hand laparoscopic scissors was used to divide the cystic duct. The motion trajectory data for this tissue division task was subsequently used in the modelling process. The motion trajectory of each trial was recorded in $\mathbf{X} = \{X_{t,i}, \mathbf{X}_{s,i}\}$ format, where the spatial data $\mathbf{X}_{s,i}$ consisted of $\{X_p, X_y, X_t, X_h, X_r\}$ from 5 axes, i.e. pitch, yaw, translation, handle's grasping angle and roll, respectively (See Figure 3-3 (a)). The trajectories of motion surgical instruments were sampled at 8.3 Hz.

Figure 4-3 (a) and (c) depict the trajectories of the tissue division task for the left hand and the right hand instruments, respectively. The time taken to complete each trial of the same task was different. Features from different trials appeared to overlap each other, as shown in the plot of handle's grasping angle in Figure 4-3 (a) and (c). This reduced the capability of GMM / GMR to model and extract the key feature of the motion. Figure 4-3 (b) and (d) are the motion data after the multi-dimensional DTW of the tissue division tasks, with the motion features aligned.

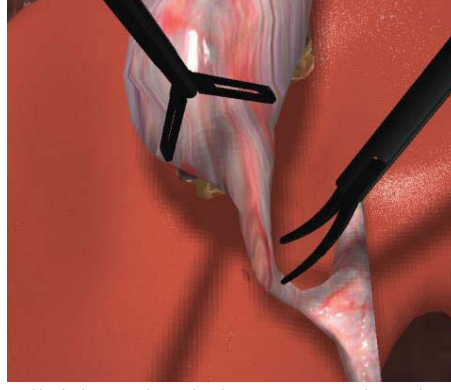


Figure 4-2 A tissue division simulation on a virtual patient in the surgical simulation system.

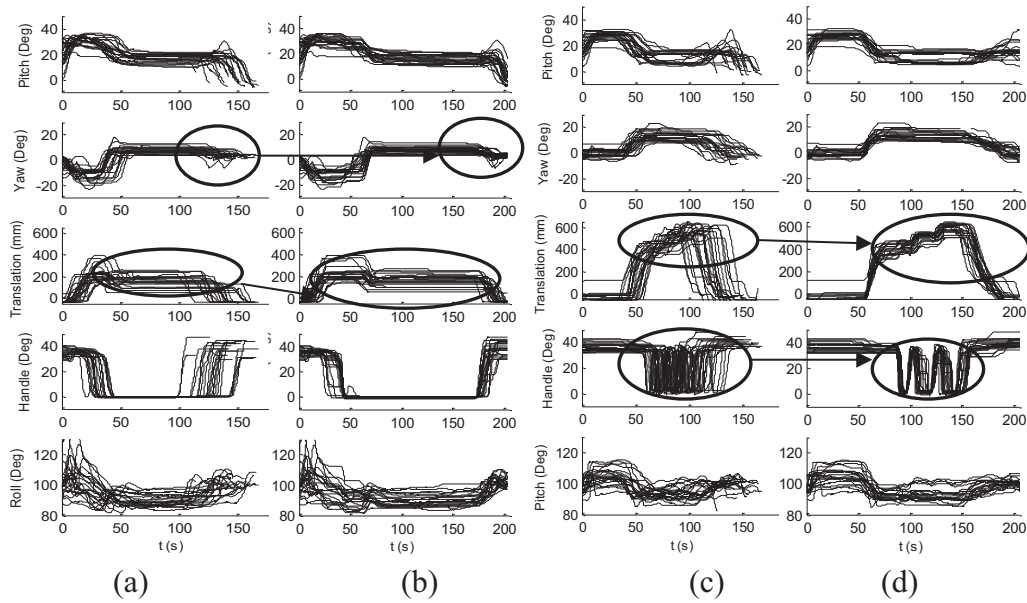


Figure 4-3 Comparison of the raw motion data collected from the simulator and the motion data after multi-dimensional Dynamic Time Warping. (a) and (c) are raw motion data, (b) and (d) are the motion data after DTW. The circled sections indicated the overlapped features in the raw motion data, and the results after DTW.

4.2.2 Results

In order to obtain the principal axis of the motion data, the PCA described in Section 4.1.1 was applied, maintaining 95 percent of the variance for the motion trajectories. The initial bandwidth h_o which was obtained using plug-in-rule method based on the distance in the latent space data $\{\mathbf{x}_{s,i}\}$, is 10.97

and 9.95 for left and right instruments, respectively. The adaptive bandwidth was determined using Equation (4.5). Figure 4-4 shows the adaptive bandwidth for one of the trials. The spatial data in the latent space $\{\mathbf{x}_{s,i}\}$ was then grouped in clusters using the adaptive mean shift method described in Section 4.1.2.

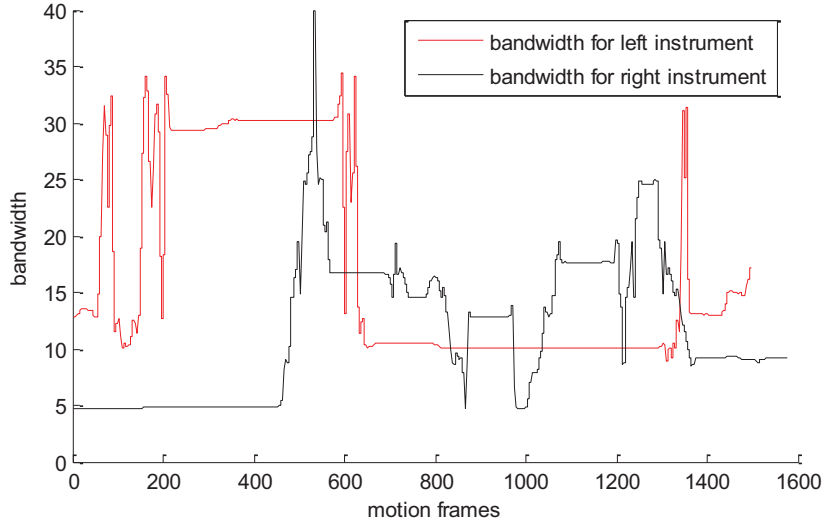


Figure 4-4 The adaptive bandwidth value for the left and right trajectories of the instrument in one demonstration.

The GMM method described in Section 4.1.3.1 was applied to model the spatial data $\{\mathbf{x}_{s,i}\}$ and the parameters of the GMM model were estimated by the Expectation Maximization algorithm [72]. Figure 4-5 (a) and (c) show the Gaussian Mixture Models trained with the motion primitives identified by the adaptive mean shift method. Eight and thirteen primitives were identified in the left and right instruments trajectories. The estimated parameters were for the data set in the latent space. For GMR regression process, they were projected back into the original space by Equation (4.9).

The GMR method described in Section 4.1.3.2 was then applied to reconstruct the trajectories in the original space. Figure 4-5 (b) and (d) show the GMR regression results of the GMM models which were trained to encode the surgical skills demonstrated. Figure 4-6 shows the 3D plot of the tissue-division task with the demonstration trajectories and the reconstructed trajectories. The implementation of the GMM and the GMR is based on a Gaussian mixture tool kit [61] available in the public domain [130].

In order to further evaluate the robustness of the proposed method, the method was applied to model a surgical task of deploying a clip with laparoscopic instruments in laparoscopic cholecystectomy using the system described in Section 3.1. In the experiment, the left instrument was used to grab and hold the gallbladder, while the right instrument approached the cystic duct and deployed a clip. The surgical task was carried out by Subject 2 and Subject 3 with the same virtual patient setup. Each subject repeated the task a number of times. Twenty two and twenty four trajectories were recorded from Subjects 2 and 3, respectively. Figure 4-7 and Figure 4-8 show the raw motion trajectory data and the mean reconstructed model of Subjects 2 and 3, respectively. Comparing the mean reconstructed model of each subject's left instrument, it is noticed that each subject manipulated the instruments differently; Subject 2 tends to focus on controlling the span of instrument swing more closely than that of Subject 3.

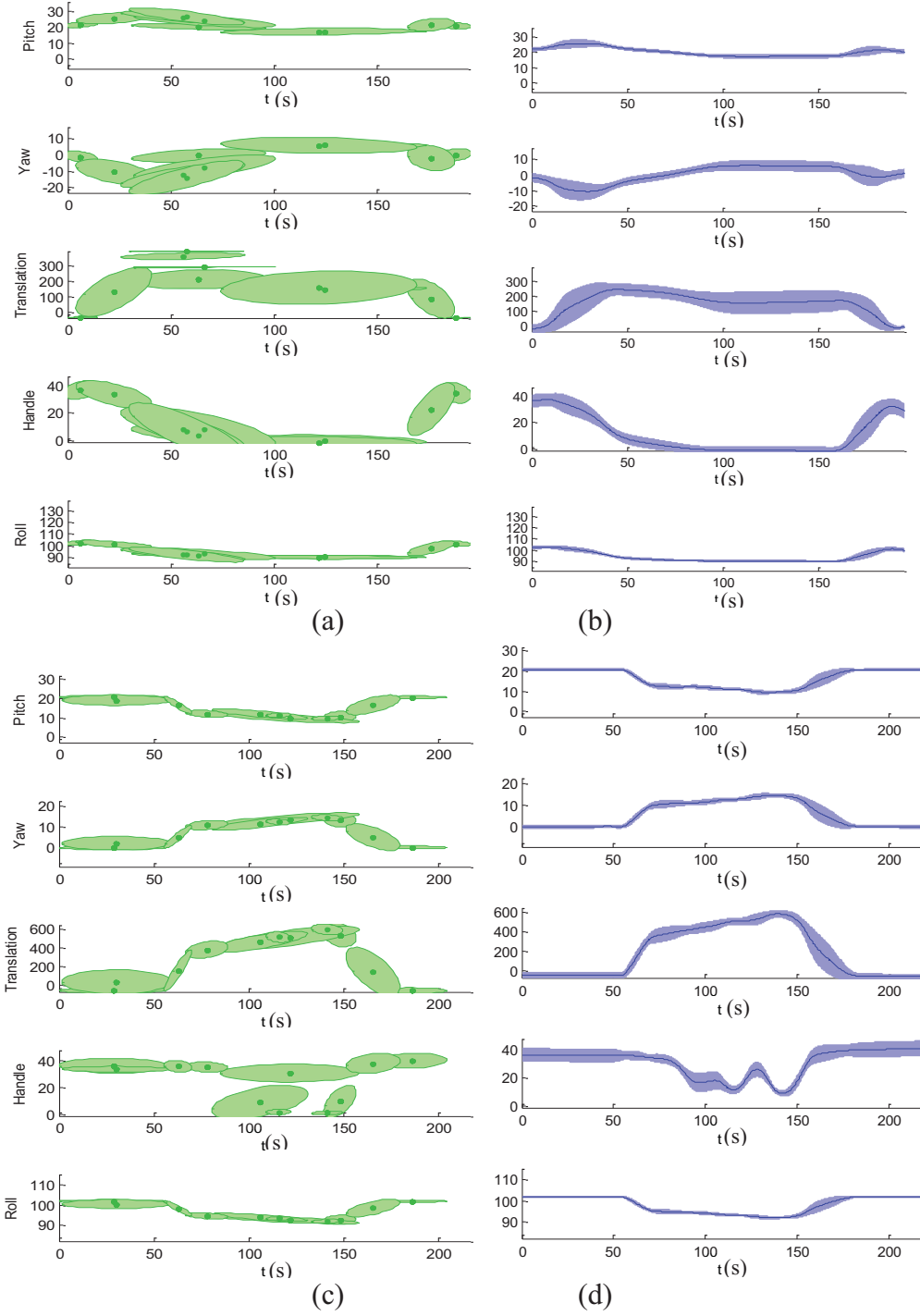


Figure 4-5 The GMM modelling and the GMR regression results based on the proposed method. (a) and (c) are the GMM encoding for the tissue division task of the left and right instruments, respectively, based on the adaptive mean shift clustering results. The spot is the mean of each Gaussian component, and the patch is the square root of covariance matrix of the corresponding Gaussian component. (b) and (d) are the GMR regression results, the solid line is the expected mean of each Gaussian model at the given time t , and the patch is the expected square root of the covariance matrix at the given time t .

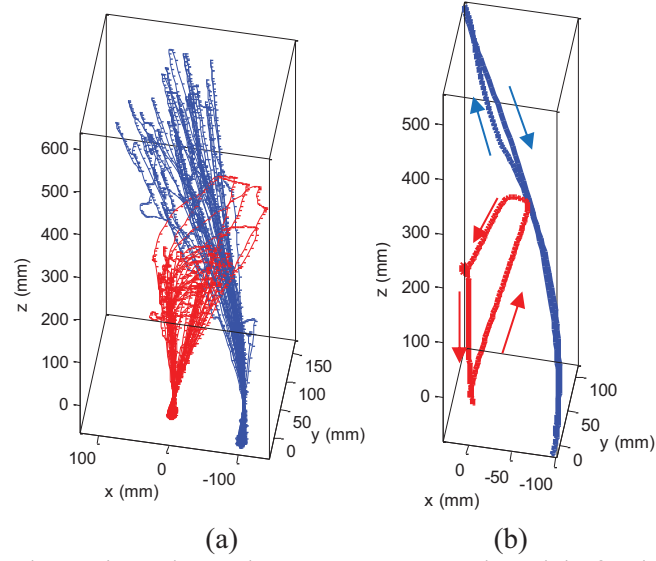


Figure 4-6 Raw motion trajectories and mean reconstructed model of Subject 1: (a) 22 motion trajectories (positional only) of the surgical tool tip in the tissue division task, (b) reconstructed mean trajectory by GMM and GMR. The orientation of instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument. The arrows indicate the direction of motion.

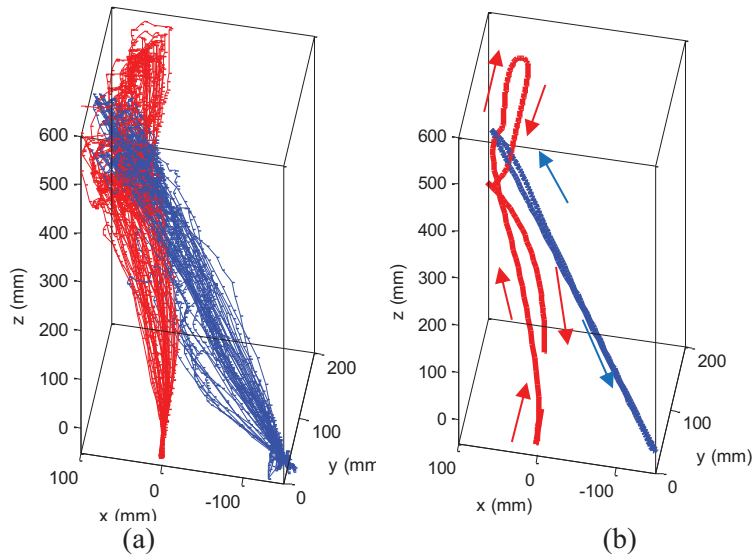


Figure 4-7 Raw motion trajectories and mean reconstructed model of Subject 2. (a) 22 motion trajectories (positional only) of the surgical tool tip in the clip deployment task. (b) Reconstructed mean trajectory by GMM and GMR. The orientation of instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument.

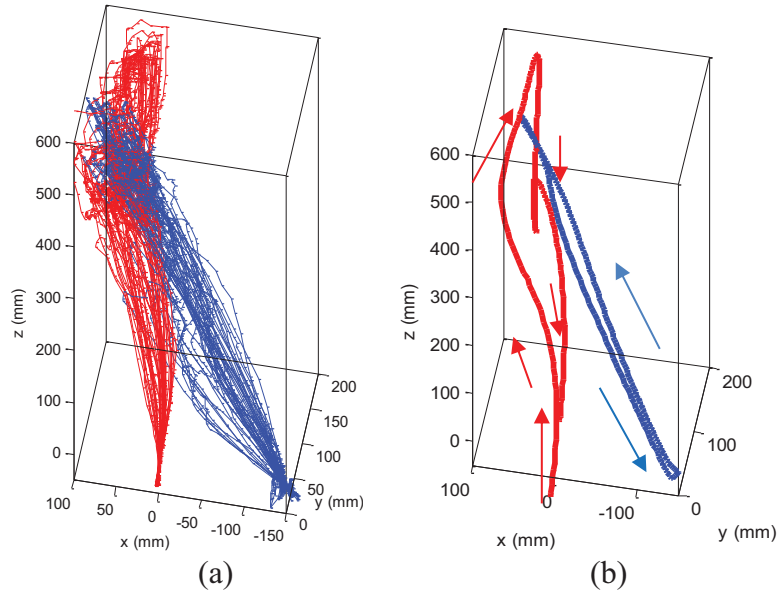


Figure 4-8 Raw motion trajectories and mean reconstructed model of Subject 3: (a) 24 motion trajectories (positional only) of the surgical tool tip in the clip deployment task, (b) reconstructed mean trajectory by GMM and GMR. The orientation of instruments and open angle of the handles are not reflected in this plot. The plot in red represents the positional information of the left instrument, and the plot in blue represents that of the right instrument.

4.3 Discussion

The adaptive mean shift method was applied to identify the motion primitives in the study. The adaptive mean shift method provides an intuitive way in determining the number of motion primitives based on the initial bandwidth which was obtained by the plug-in-rule method [124]. However, the performance of the adaptive mean shift method relies on its initial bandwidth and the adaptive bandwidth function [126]. It can be obtained by several methods, such as plug-in-rule, iterative, maximal smoothing, biased cross-validation, least square cross-validation and Asymptotic MISE approximation [125]. The bandwidths obtained from different method vary slightly. Smaller bandwidth is preferred to avoid missing tiny features in the data.

Root Mean Square (RMS) error was applied to evaluate the quality of the motion model through the reconstructed motion model [65]. RMS error of the reconstructed trajectory with respect to the demonstrated trajectory after DTW were calculated as follows

$$RMS = \sqrt{\frac{1}{MN} \sum_{j=1}^M \sum_{i=1}^N (\mathbf{X}_{s,i} - \mathbf{X}_{s,i})^2}, \quad (4.14)$$

where M is the number of trials, N is the number of observations in each trial, \mathbf{X}_e and \mathbf{X}_s are the expected spatial components and the spatial components from the demonstrations, respectively.

The quality of the motion model obtained based on different methods in identifying the motion primitives were compared, i.e. adaptive bandwidth mean shift method, fixed bandwidth means shift method and K-means method. The number of primitives required for K-means method was determined from the tests using adaptive means shift method. Figure 4-9 (a)-(d) show the GMM modelling results based on K-means method and fixed bandwidth mean shift clustering method. Comparing with Figure 4-5, it is noticed that adaptive mean shift-based method captured motion primitives with more focused Gaussian components than that of K-means and fixed bandwidth-based methods. The fixed bandwidth mean shift identified 6 and 14 primitives from the motion trajectory of left and right instruments, respectively. Although the three methods employed similar number of motion primitives, Table 4-1 shows the GMMs with adaptive mean shift method produces smaller RMS error comparing with that of the K-means method and fixed bandwidth methods. In addition to the smaller errors compared to other methods, the Adaptive mean shift method is preferred because the Adaptive mean shift-based method

begins the motion primitive identification from the trajectory data. This enables the entire modeling process to require minimal user intervention. In contrast, K-means-based method requires the user to manually specify the number of motion primitives contained in the data.

The adaptive mean shift method also showed advantages in preserving the dexterous features in motion. For example, the handle motion (Figure 4-3 (d)) showed several open and close actions. These features have been encoded and reconstructed by the GMM / GMR with adaptive mean shift method, as shown in Figure 4-5 (c) and (d). However, these features were not captured in GMM with K-means and fixed bandwidth methods (Figure 4-9 (b) and (d)), even the number of primitives used for K-means method was the same as the number of primitives obtained by the adaptive mean shift method, and the fixed bandwidth method obtained similar number of primitives with the adaptive mean shift method.

Another advantage of the Gaussian mixture modelling method based on the adaptive mean shift method is that it does not need to specify the number of the Gaussian components required to model the demonstration. While it is possible to have a better fit of the trajectory with a high number of Gaussian components, this will be at the expense of poor generalization capability and potential risks of over fitting. However, the adaptive mean shift-based method required an initial bandwidth to be determined in pre-processing. A tool kit from [127] took about 5-6 minutes on a PC with Intel i7-2653QM 2.0 GHz processor to calculate the bandwidth of a vector with 8538 rows.

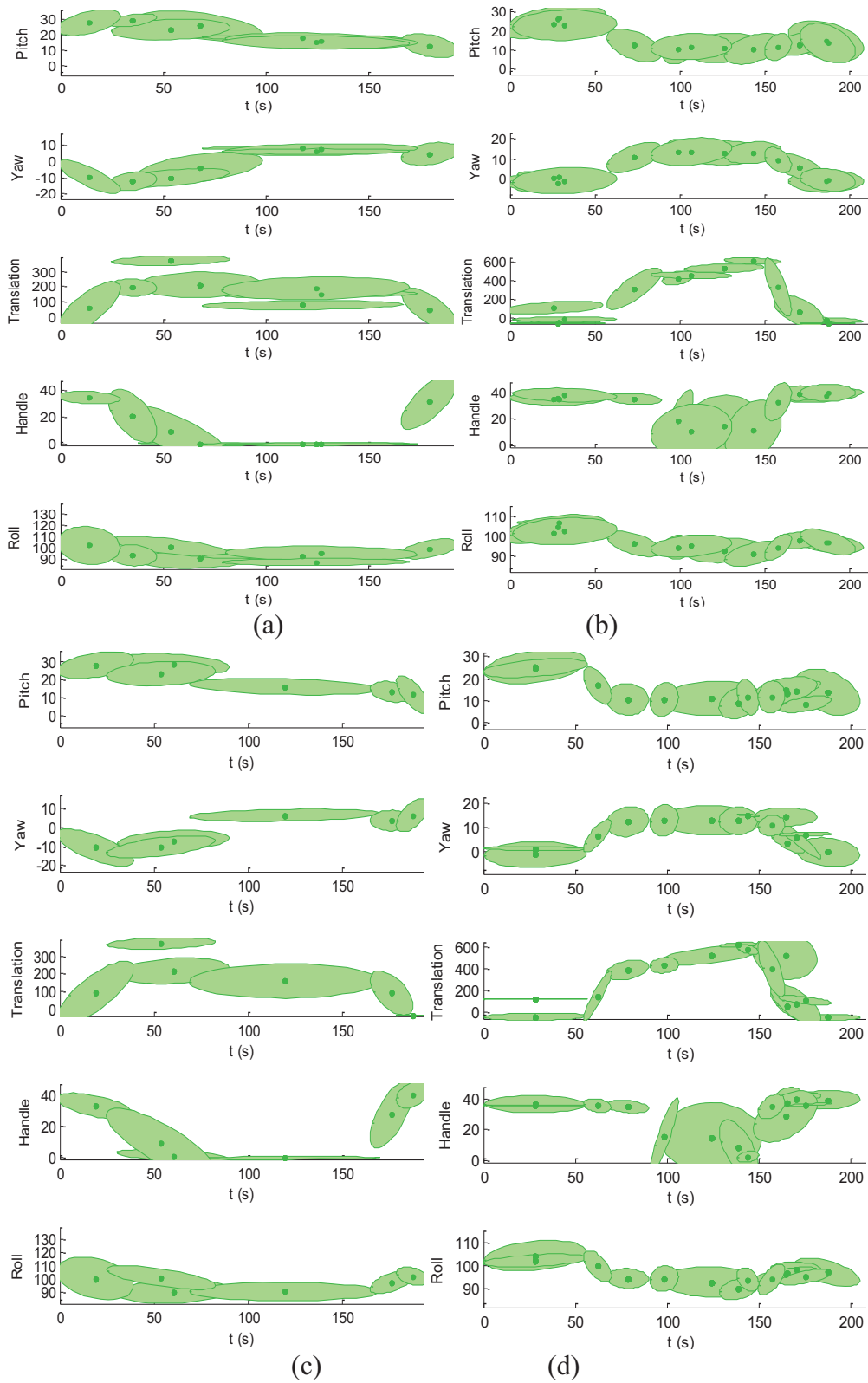


Figure 4-9 GMM modelling based on K-means method and fixed bandwidth mean shift method. (a) and (b) are the GMM modelling results with K-means clustering method for the left and right instruments trajectories, respectively. (c) and (d) are the GMM modelling results with fixed bandwidth clustering method for the left and right instruments trajectories, respectively.

Table 4-1 The RMS error of the rotational joints of the reconstructed trajectory to the demonstrations after DTW.

	Left Tool Trajectory (Degree)	Right Tool Trajectory (Degree)
Fixed bandwidth mean shift	± 3.13	± 3.19
K-means	± 3.08	± 3.66
Adaptive mean shift	± 3.05	± 3.08

PCA is necessary in the analysis of the motion trajectory data. The PCA can be applied for reducing the dimensionality and the noise, and also to rotate the data to the axis that allows the clustering algorithm to identify the motion primitives effectively. When dimensionality reduction is not required for the data set, PCA is necessary to rotate the data set according to the eigenvector of covariance matrix of the data set, and to align the data in its principal axis. The tissue division trajectories were applied with the adaptive mean shift method directly without PCA and the identified motion primitives were used to train the GMM models. Figure 4-10 shows trained GMM modelling and GMR regression results. The data across large time spans were grouped in the same motion primitive which significantly reduced the capability of the Gaussian Mixture Regression model. Table 4-2 shows that the RMS error of the reconstructed trajectory from the demonstrations without PCA analysis is larger than that with the PCA analysis. Therefore, the PCA is an important component of the solution.

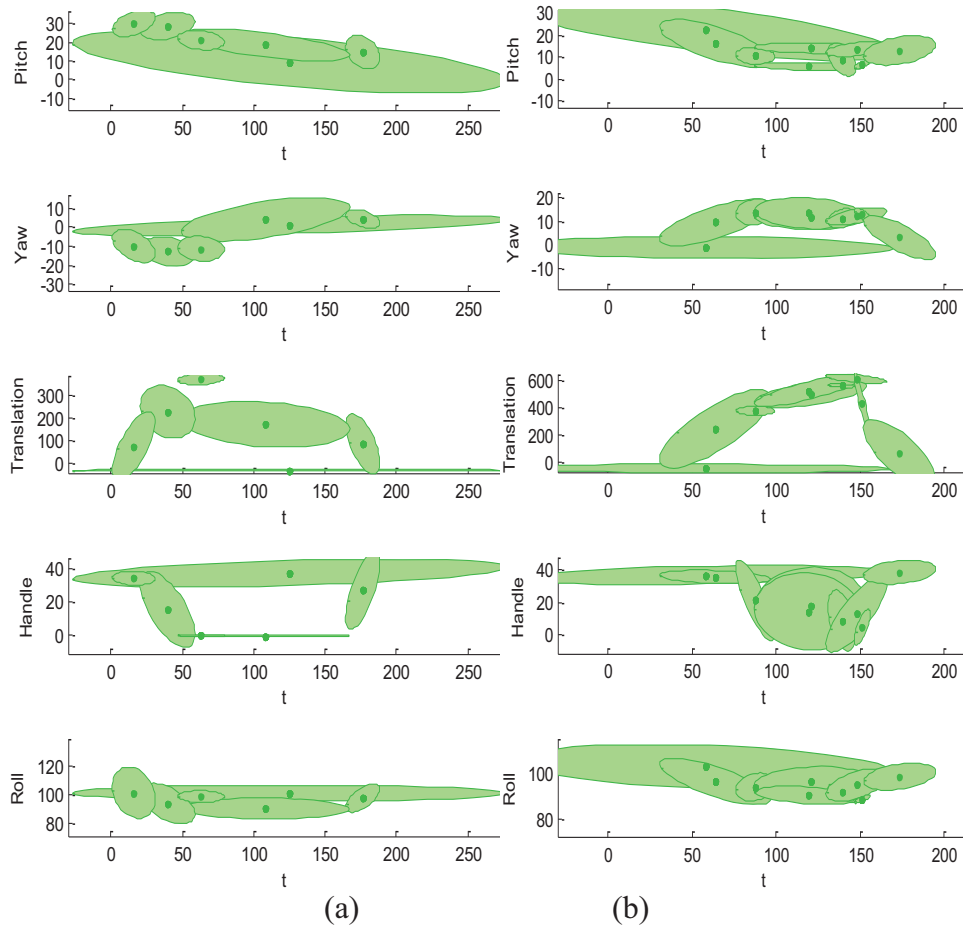


Figure 4-10 The GMM modelling results of the tissue division trajectory without the PCA analysis. The data across large time span were grouped in same motion primitive. (a) and (b) are the GMM modelling of trajectories for left and right instruments, respectively.

Table 4-2 Effect of PCA on RMS error of rotational joints of the reconstructed trajectory to the demonstrations after DTW.

	Left Tool Trajectory (Degree)	Right Tool Trajectory (Degree)
With PCA Analysis	± 3.05	± 3.08
Without PCA Analysis	± 3.14	± 3.77

Our approach is suitable for modelling of surgical skills with a specific sequence of motion primitives, such as the division and clipping tasks modelled in this study. Both tasks require grabbing and holding onto the object first, before performing the task at certain locations. While performing the task, the pattern of opening and closing of instrument handle is consistent among

the user's executions. Clear motion sequences can be identified from the user's demonstration. Surgical suturing could also be modelled by the proposed method, as it required both hands to conduct the motion in sequences. Surgical operations in which the sequence of motion is not critical, may not be represented by GMM effectively. The proposed method focuses on extraction and reconstruction of a generic model from demonstrations conducted by the user. It does not include the collaboration between two instruments and tool-tissue interaction. In order to consider these factors, the velocity of each instrument and the deformation of organ or tissue have to be modelled. Although the robustness of our method was evaluated with different surgical tasks, the study was still limited by the size of sample, the complexity and the range of surgical procedures. With the development of the surgical simulator in our experiment, more surgical procedures could be studied in the future to demonstrate the generalizability of the proposed method.

4.4 Summary

Learning from experienced surgeons is an efficient way of transferring surgical skills from the experienced surgeons to the novice surgeon. The method of robotic learning by demonstration is an approach to model the surgical skills and facilitate it for surgical training from the perspective of motion trajectory. The trained motion model in learning by demonstration approach can serve as a generic model representing surgical skills. The motion model can then be used by the robots to provide guidance to the novice surgeon. Experimentation of our robotic surgical training system and the underlying technology with novice surgeon is ongoing.

The method proposed in this study demonstrates the feasibility of modelling skills without specifying number of motion primitives. This has contributed to the robustness of our robotic surgical training system. Adaptive mean shift method has been applied to identify the motion primitives, and the Gaussian Mixture Models is trained by demonstrations to represent a surgical skill. However, collaboration among multiple instruments is essential in the execution of many surgical tasks. Developing collaborative models to represent the cooperation of multiple surgical instruments is required. The various spatial and temporal constraints in surgery also have to be taken into consideration for a complete simulation of a surgical operation. For example, in situations where a certain location / obstruction has to be avoided, or a specific location that must be passed through in order to reach the targeted site, constraints dependent on individual patient anatomy have to be considered.

5 MOTION INTENTION RECOGNITION AND ITS APPLICATION IN SURGICAL TRAINING

A user could benefit from working on a task with a robot if the robot could perceive the needs of the user. Therefore, understanding the intention of the trainee or surgeon in a surgery is a critical step to bring the robot-assisted technology into surgical training or real surgery. The robotic surgical trainer needs the capacity of recognizing the motion intention of the surgeon. In this chapter, Hidden Markov Models (HMM) was applied to study the motion intention based on the trajectory of the surgical instruments in a laparoscopic surgery. A stacked HMM method is proposed to recognize the motion intention at different levels. Observation feature is the key component in successful HMM application. There are two approaches to determine the observation feature for the recognition problem [100]. One approach is to use the raw data as the input, it is named 'template' in [100]. Such 'template' includes coordinate points of the trajectory path. It is one of the major feature representations in gesture recognition. The drawback of using 'template' data as feature is that the recognition is not robust within a given class [100]. The other approach is to use the extracted important features, such as features in frequency information. The advantage of using features from the frequency domain is the stability of recognition. The effect of both types of observation features are studied in this chapter.

This chapter is arranged as follows: Section 5.1 introduces the stacked HMMs for motion intention recognition; Section 5.2 describes the configuration of the HMMs in the stacked structure; Section 5.3.1 describes

the experiment design and surgical simulation; Section 5.3.2 discusses the performance of the stacked HMMs and the work is summarized in Section 5.4.

5.1 Stacked Hidden Markov Models

5.1.1 HMM for Motion Intention Recognition

The target of motion intention recognition process is to identify the motion fragment which is represented by an observation sequence O . This is achieved by identifying the HMM with the highest probability in a recognition model with the given motion fragment, i.e.

$$g = \arg \max_{all \lambda} P(\lambda | O), \quad (5.1)$$

where λ is a HMM trained to represent the motion intention, and O is the observation sequence to be recognized. Hence this process relies on the computation of $P(\lambda | O)$. From the Bayes formula, the posteriori probability given the observation sequence can be written as

$$P(\lambda | O) = \frac{P(O | \lambda)P(\lambda)}{P(O)}. \quad (5.2)$$

In Equation (5.2), $P(O)$ is a constant for a given observation sequence, $P(\lambda)$ is the probability of the motion primitive which characterizes the likely sequence of motion primitive described by the features in the observation sequences. The probability of the observation sequence generated by an HMM, $P(O | \lambda)$ is the only variable in Equation (5.2). It can be obtained by the forward-backward algorithm. The problem of intention recognition can then be simplified as

$$Intention = \arg \max_{all \lambda} P(O | \lambda), \quad (5.3)$$

where λ is the HMM trained to represent different intentions.

5.1.2 Stacked Hidden Markov Models

In a laparoscopic surgery, each hand performs a specific role, and both hands collaborate to accomplish the task. With this observation, a stacked HMMs was proposed, as illustrated in Figure 5-1, to model the motion intention from instruments trajectories. The structures shown in Figure 5-1 can be extended to multi-layer stacked HMMs. There are multiple groups of HMMs in each layer running in parallel. Each group of HMMs is a recognition model. Each recognition model is formed multiple HMMs. Each HMM represents one particular intention in the recognition model. The output from each HMM can be taken as one of the observation features for the next level of recognition models, or a winner-takes-all strategy can also be used to generate observation features for the next level of recognition models.

The advantage of the stacked HMMs is that it reflects the intention of the individuals in a common task group as well as reflects the relationship on the collaboration among the individuals. For example, there are multiple individuals working together to achieve a simple goal, and multiple groups of such individuals working together in achieving a common goal. The stacked HMMs can be applied to represent the collaboration among the individuals and the groups. In the first level, each individual's intention is recognized by a recognition model which consists of numbers of possible intentions. Different individuals at the same layer may have different intentions, hence there are multiple recognition models running in parallel to recognize the intentions of each individual. Each recognition model is fed with the observation sequences from the sensors or features extracted from the sensed data. The HMM which

produces the highest likelihood represents the intention of the observation. The output of the recognition model from different individuals at layer one are combined and formed as an observation feature and fed to the recognition models at the next layer. Such combination can be formed with different format as suggested by the collaboration relationship among the individuals, such as the weight of the individual in the group. In the next layer, the recognition model takes the observation sequences and recognizes the intention of the group. The recognition model formed by the HMMs at different layers are utilized to recognize the intentions at different levels. The frequency of changing intentions at different levels is different. Therefore, recognition frequency at different layers can be different.

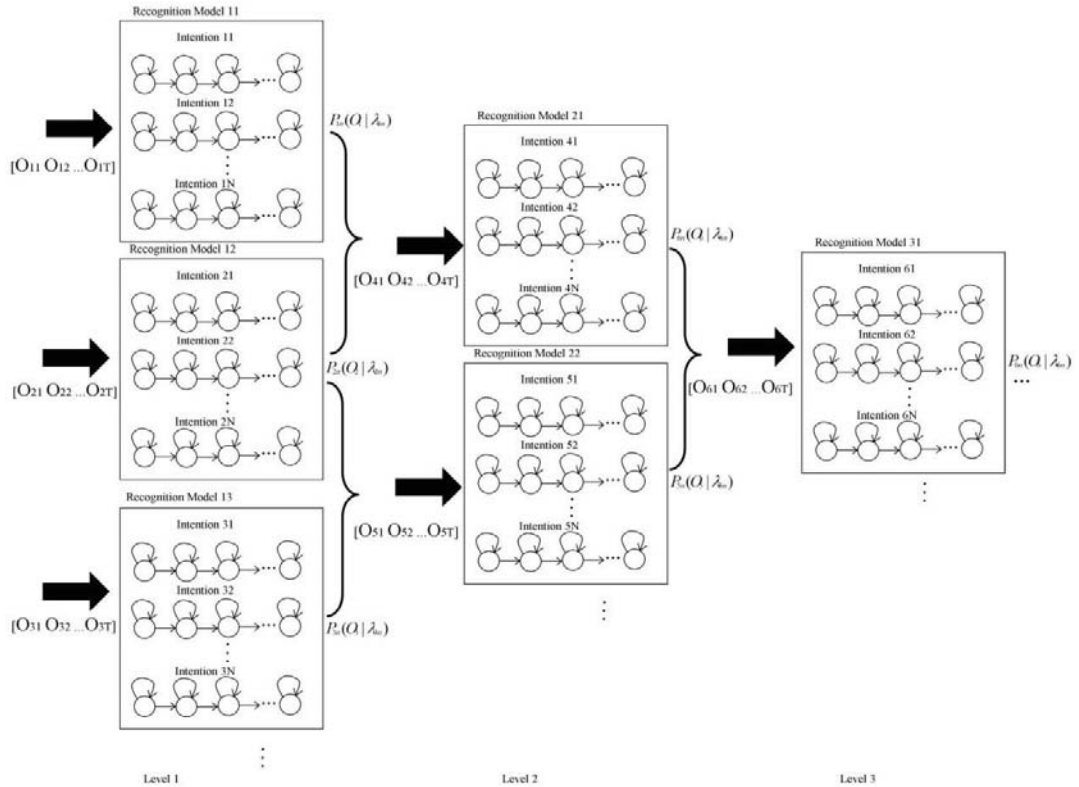


Figure 5-1 Stacked HMMs for motion intention recognition.

5.2 Stacked HMM for Laparoscopic Surgical Training

A two layer stacked HMM, as shown in Figure 5-2, is configured for motion intention recognition in laparoscopic surgical training. Layer 1 is named as primitive layer, and the recognition model is called the primitive recognition model. Layer 2 is named as subtask layer and the associated recognition model is called the subtask recognition model. There are two recognition models in the primitive layer for the left and right instruments, respectively. They are respectively denoted as λ_L and λ_R , each recognition model consists of k HMMs required to represent the motion, or in other words, the number of motion intentions. Note that k could be different for left and right instruments.

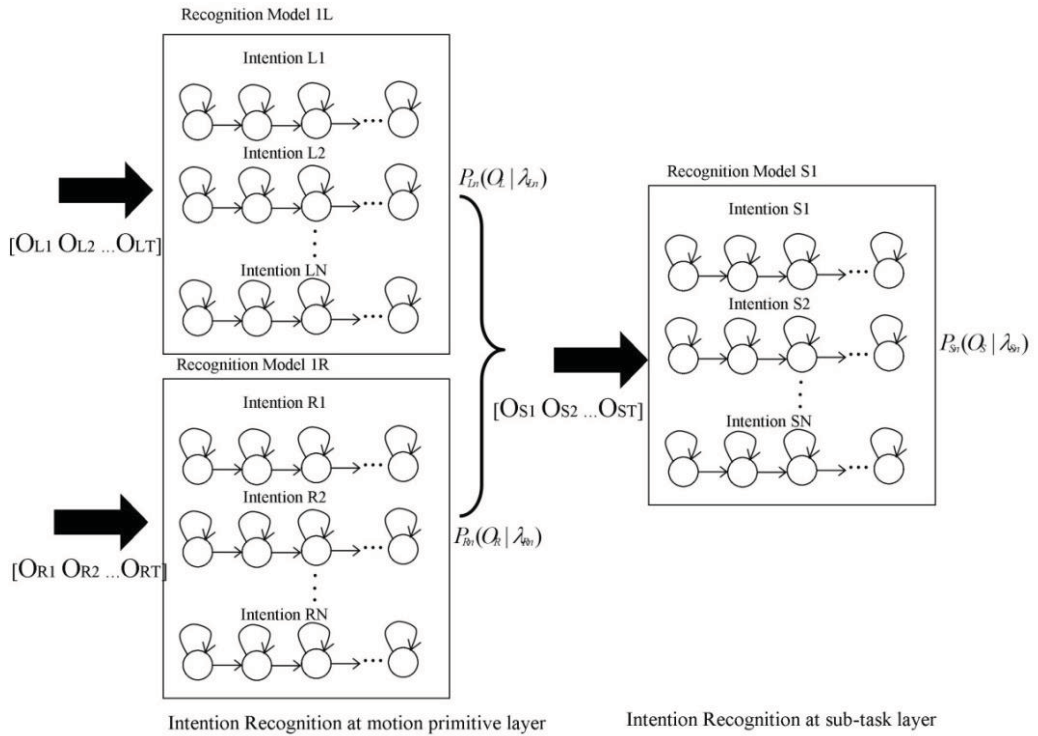


Figure 5-2 Stacked HMMs for motion intention recognition in laparoscopic surgical training.

The motion intention is determined by identifying the maximum likelihood produced by the HMMs in the respective recognition model. The subtask recognition model λ_{sub} in subtask layer is used to recognize the subtasks, where the subscript sub is the number of HMMs in the subtask recognition model. It corresponds to the number of subtasks.

5.2.1 Observation Features for the HMMs

The observation features in this research are extracted from a virtual surgery described in Section 5.3.1. They can be expressed in the spatial domain and the frequency domain for the primitive recognition model. Observation features in the spatial domain includes: position and orientation of tool tip relevant to a specific point on the organ, opened angle of grasping handle expressed by p ; velocity and acceleration of instrument, including the velocity and acceleration of grasping handle, expressed in v and α , respectively. The individual elements in p , v and α are not express in the same measuring units: velocity of tool tip is in m/s, the velocity of grasping handle is in rad/s. They are normalized to remove the weight effect induced by the measuring units. Let O be the vector representing the information of each motion frame, it is expressed as

$$O = \{p \quad v \quad \alpha\}. \quad (5.4)$$

In addition to using the features in the spatial domain, the intention recognition using features extracted in the frequency domain were also studied. Power Spectral Density (PSD) is taken as the observation features in this research. The human motion frequency bandwidth is narrow, therefore 16 point Fast Fourier Transformation (FFT) is performed on the observation

sequences described by Equation (5.4). The result of the FFT is expressed by the complex vector $\mathbf{Z}_n = [z_0 \ z_1 \ \dots \ z_{15}]_n$, where n is the number of observations in the trial. The PSD is expressed as

$$P(\vec{Z}) = [P_0(\vec{Z}) \ P_1(\vec{Z}) \ \dots \ P_{k/2}(\vec{Z})], \quad (5.5)$$

where $k = 16$, $P_0(\vec{Z}) = \|z_0\| / H_{ss}$; $P_i(\vec{Z}) = (\|z_i\| + \|z_{k-i}\|) / H_{ss}$, $i = 1 \dots 7$;

$P_{k/2}(\vec{Z}) = \|z_{k/2}\| / H_{ss}$; $H_{ss} = k \sum_{i=1}^k H_k^2$. H_k is a hamming window of width 16.

The recognition results from the primitive layer are fed to the subtask layer to recognize the motion intention in the subtask level. The features fed to the subtask layer could be either the recognized motion primitive which is the index of the recognized HMM from the primitive recognition model, or the observation probability $P(O|\lambda)$ from the primitive recognition model. The normalized observation probabilities P_L and P_R of both primitive recognition models are combined as an observation feature for the subtask recognition model λ_{sub} . However, observation probabilities P_L and P_R with similar magnitude do not necessary relate to different motion primitives because the observation probabilities P_L and P_R are obtained from different primitive recognition model which are trained by the features obtained from the left and right instrument, respectively. In order to make the observation probabilities P_L and P_R distinguishable when combining them together, the observation probabilities produced by primitive recognition model λ_R are multiplied with a constant factor of -1. The observation feature for the subtask layer is expressed as

$$O_{sub} = \{P_L \quad -P_R\}, \quad (5.6)$$

where $P_L = \{P_{L,1} \quad P_{L,2} \quad \dots \quad P_{L,k}\}$, $P_R = \{P_{R,1} \quad P_{R,2} \quad \dots \quad P_{R,k}\}$, and k is the number of HMMs in the primitive recognition model.

5.2.2 HMM Configuration

Continuous HMM with Gaussian distribution are applied to construct the recognition models. The states in the primitive recognition models represent the 'mental state' of the subject which might relate to motor imagery. Motor imagery is a mental process that an individual rehearses a required task. The subject may apply identical mental state at different stages during the execution of a task. Therefore, full transition pattern is applied to model state transition probability. The states in the subtask recognition model represent the motion primitives. Although the two primitives in a motion trajectory might be very similar in certain means such as moving direction and speed, they are still distinguishable when they are considered in the scope of hidden Markov process. The primitives prior to them make them distinguishable from each other.

5.2.3 HMM Training and Recognition

Data collected from a virtual surgery for both instruments denoted by O_L and O_R are classified into k clusters each using K-means algorithm. The number of clusters is determined empirically. If too many clusters are specified, the clustering algorithm may not be able to converge or produce an empty cluster. Each cluster is taken as one motion primitive. Training and testing data to the recognition models are obtained by continuously taking

windows of the data from the motion primitives $O_{L,k}$ and $O_{R,k}$ in the k^{th} cluster, as illustrated in Figure 5-3. The width of the window can be varied. The k^{th} HMM in primitive recognition model $\lambda_{L,k}$ and $\lambda_{R,k}$ were trained with observation sequence obtained directly from the sensors. The subtask recognition model λ_{sub} are trained by the observation features formed by Equation (5.6) with the whole subtask.

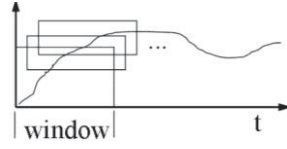


Figure 5-3 Observation sequence formed by data windows.

5.3 Experiments

Experiments were conducted to verify the effectiveness of the proposed motion intention recognition model. In the following subsections, the virtual surgery used for the experiment is presented first, and then the experimental results on motion intention recognition with different model configurations.

5.3.1 Surgical Simulation and Experiment Design

Surgery is a systematic process. The entire surgical procedure can always be divided into several subtasks in pre-surgical planning. Each subtask consists of numerous motion primitives. In this study, the surgical task described in Section 3.1 was executed on the IRAS training system. Data were acquired at 14 Hz by the system. Collected data were later re-sampled to 42 Hz. The motion intention of experimental subjects were studied based on the motion trajectories of the virtual surgical procedure.

A segment of the cholecystectomy surgical procedure was selected as the experimental scenario in the simulation. This segment begins with the liver and the gallbladder lifted up and exposed. A grasping forceps (Figure 5-4 (a)) is inserted from the left port to grasp the Hartman's pouch of the gallbladder and pulled to stretch the cystic duct. A small hook electrode (Figure 5-4 (a)) is inserted from the right port to ablate the connective tissue and dissect the cystic duct. When the ablation process is completed, the instrument in the right port is changed to a curved forceps (Figure 5-4 (b)). This forceps is inserted between the cystic duct and the liver for inspection to ensure that all connective tissue has been fully ablated. A clip applicator (Figure 5-4 (c)) is then inserted from the right port to deploy three clips onto the cystic duct. While the clips can also be deployed on the cystic artery in real surgeries, the artery is not modelled in this scenario. After deployment of the clips, the instrument in the right port is changed to a laparoscopic scissors (Figure 5-4 (d)), and the cystic duct is divided. Two clips are left on the cystic duct to ensure that the cystic duct has been clamped securely.

Special requirements from medical perspective such as the location of grabbing, the orientation of curved forceps, need to be taken into consideration while using each instrument. During the entire virtual surgery, the main task of the instrument in the left port is to grab onto the gallbladder and stretch the cystic duct, providing room and allowing the instrument from right port with more access to carry out the procedure. The procedure described above was chosen as the evaluation procedure as it is a critical procedure in cholecystectomy surgery. Collaboration of the two instruments, and tool tissue interaction appeared in these simple tasks. The stacked structure takes into

consideration the individual instrument motion and the collaboration of instruments as well. The entire series of tasks can be further divided into 4 subtasks based on the instrument in the right port as follows:

Subtask 1: ablation of the connective tissue and dissection of the cystic duct;

Subtask 2 : checking the clearance between the cystic duct and the liver;

Subtask 3 : deployment of three clips on the cystic duct; and

Subtask 4 : division of the cystic duct.

Each subtask was modeled by a state flow diagram as shown in Figure 5-5.

They can be represented using 6-7 states.

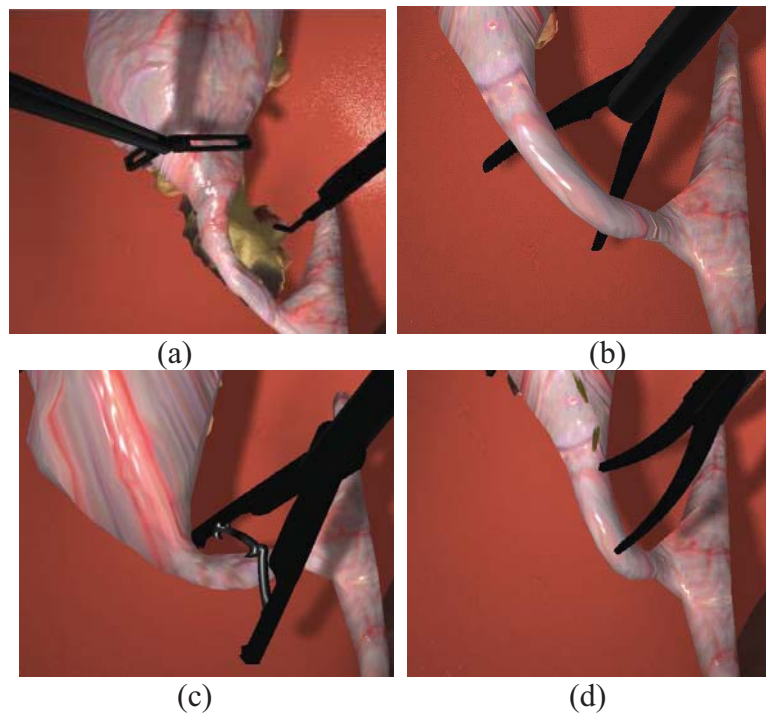
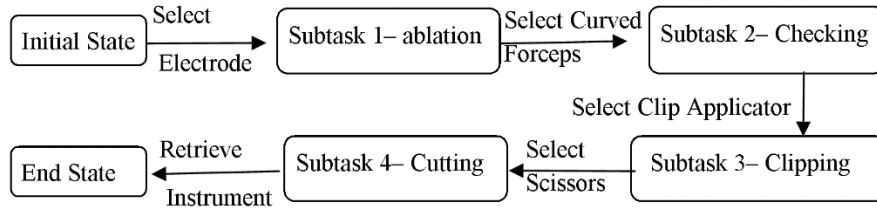
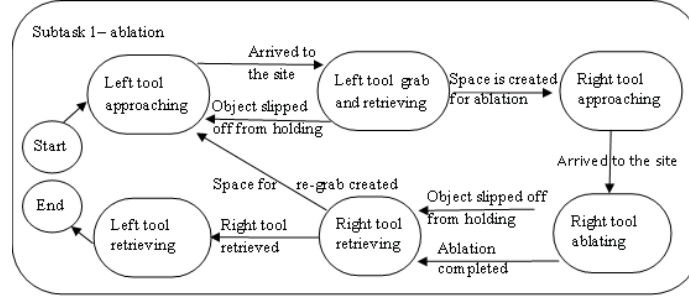


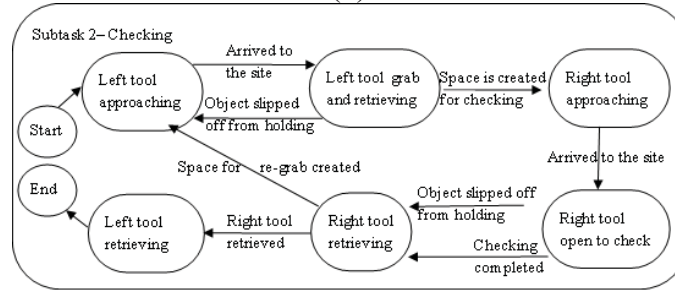
Figure 5-4 (a) Grasping forceps and hook, (b) Curved forceps, (c) Clip applicator, (d) Scissors.



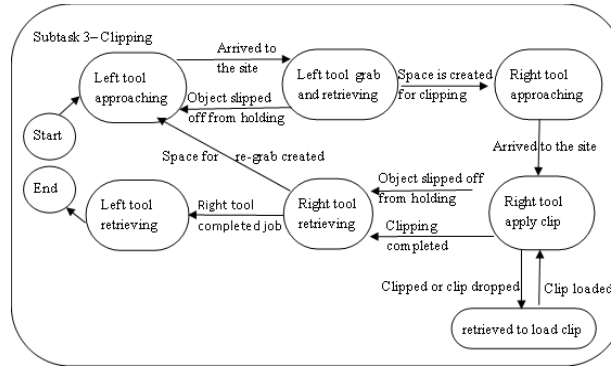
(a)



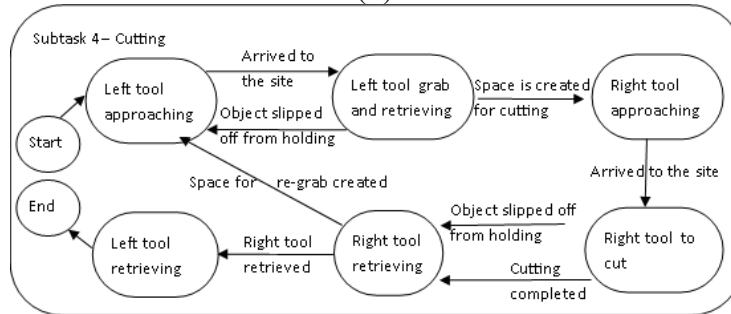
(b)



(c)



(d)



(e)

Figure 5-5 (a): State diagram for the surgical procedure; (b) (c) (d) (e): state diagrams for the respective sub tasks in Section 5.3.1.

5.3.2 Experiment Evaluation and Discussion

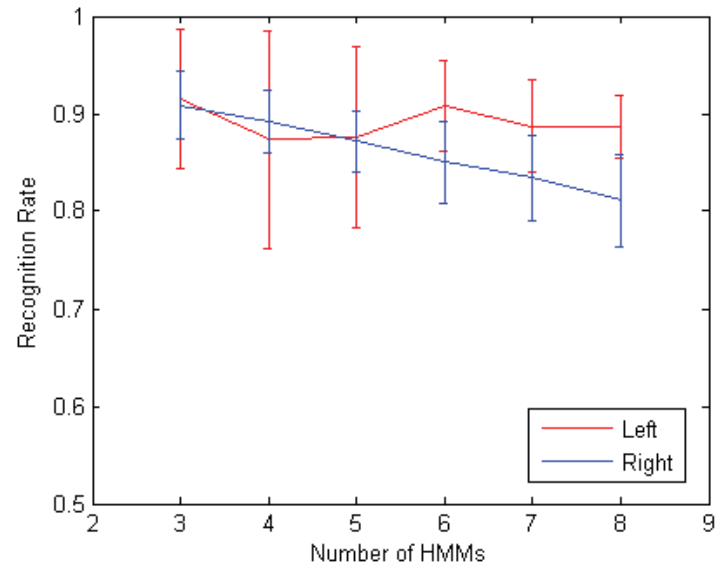
Twelve subjects, with an average age of 22.3 ± 3.1 years participated in a study to evaluate the proposed motion intention recognition method. Each subject performed the virtual surgery described in Section 5.3.1 for 10 repetitions. A total of 120 trajectories were collected. Sixty trajectories were randomly chosen for training the HMMs, the rest of the 60 trajectories were used for testing purposes. In the following subsection, the effect of various HMM configurations to the recognition rate are presented, such as the number of HMMs applied to construct the recognition models, the number of states in a HMM, and the number of Gaussian components applied to model the training and testing data.

5.3.2.1 Primitive Layer

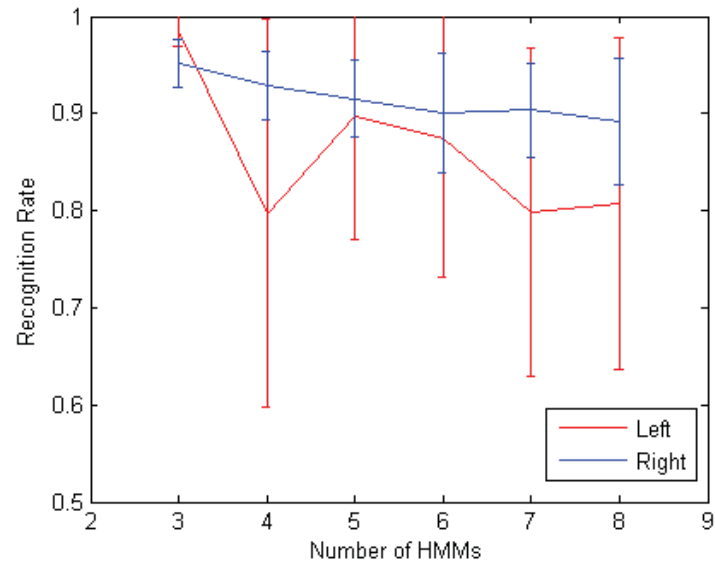
Three set of recognition tests was performed as an initial study to investigate the effect of

1. Number of HMMs in the recognition model;
2. Number of Gaussian components in representing the data; and
3. Number of states in constructing a HMM in the recognition model.

Figure 5-6 (a) and (b) show the effects of the number of HMMs used to construct primitive recognition model in the frequency and spatial domain, respectively. The recognition rate decreases as the number of HMM increases. When the virtual surgical procedure is represented by a small number of intentions, the difficulty level of distinguishing the intentions is lower, hence the recognition rate is higher.



(a)



(b)

Figure 5-6 Effects of HMM numbers to recognition rate in primitive recognition model. HMMs were configured with 3 states, with 3 Gaussian components for observation sequence. (a) Recognition rates in the frequency domain. (b) Recognition rates in the spatial domain.

The recognition model was tested by modelling the data with different number of Gaussian components. Table 5-1 and Table 5-2 are the recognition rates in the frequency domain and spatial domain, respectively. The experimental results show the number of Gaussian components has no significant effect on recognition rate.

Table 5-1 Recognition rates of the primitive recognition model in the frequency domain. Training and test data were represented by different number of Gaussian components. Three HMMs were applied to represent the motion intentions. Each HMM was set with 3 states. Twenty frames were taken for each observation.

Instrument	No. of Gaussian components			
	3	4	5	6
Left	91.5±7.2%	91.5±7.2%	91.5±7.2%	91.5±7.2%
Right	90.9±3.5%	90.9±3.5%	90.9±3.5%	90.9±3.5%

Table 5-2 Recognition rates of the primitive recognition model in the spatial domain. Three HMMs were applied to construct the recognition model. Each HMM was set with 3 states. Twenty frames were taken for each observation.

Instrument	No. of Gaussian components			
	3	4	5	6
Left	88.4±7.2%	88.2±7.2%	88.6±7.2%	88.7±7.2%
Right	95.0±3.5%	95.0±3.5%	95.2±3.5%	95.3±3.5%

The number of states in a HMM represents the mental state which drives the motion. Primitive recognition models were tested with number of states from 3 to 6 in both spatial and frequency domains. Since Table 5-1 and Table 5-2 show that the number of Gaussian components has no significant effect to recognition rate when the number of Gaussian components is set to 3 for examining the effect of number of states in HMM. Table 5-3 shows that the recognition rate is consistent with respect to number of states when observation features are taken from the frequency domain. The standard deviation is also smaller than that of the spatial domain shown in Table 5-4. Experimental results have shown that the recognition rate is stable when representing the data in the frequency domain. However, pre-processing of data is required for the recognition to be conducted in frequency domain. Figure 5-7 shows an example of primitive recognition result when the features were represented by PSD. Table 5-4 shows that the recognition rate gradually

improves as the number of states increases from 3 to 6 when the observation features are represented in the spatial domain. This result is in line with the state diagram shown in Figure 5-5 that represents the process of the subtasks. Figure 5-8 shows an example of primitive recognition result when the features are represented in the spatial domain.

Table 5-3 Recognition rates of the primitive recognition model in the frequency domain. HMMs in the primitive recognition were configured with different number of states. Intentions were represented by 8 HMMs. Twenty frames were taken for each observation.

Instrument	Number of States in each HMM			
	3	4	5	6
Left	88.7±3.2%	88.7±3.2%	88.7±3.2%	88.7±3.2%
Right	81.1±4.7%	81.1±4.7%	81.1±4.7%	81.1±4.7%

Table 5-4 Recognition rates of the primitive recognition model in the spatial domain. HMMs in the primitive recognition were configured with different number of states. Intentions were represented by 8 HMMs. Twenty frames were taken for each observation.

Instrument	Number of States in each HMM			
	3	4	5	6
Left	80.7±17.1%	81.0±16.9%	81.1±16.9%	81.6±16.3%
Right	89.2±6.5%	89.5±4.9%	90.1±4.4%	90.0±4.5%

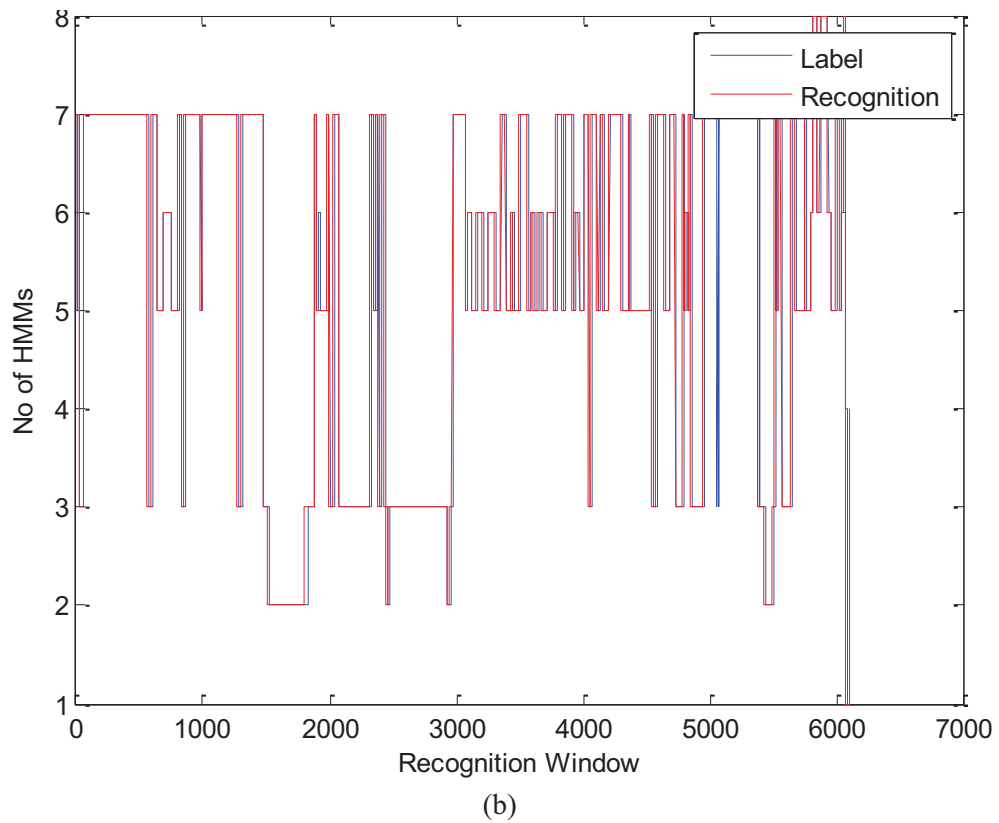
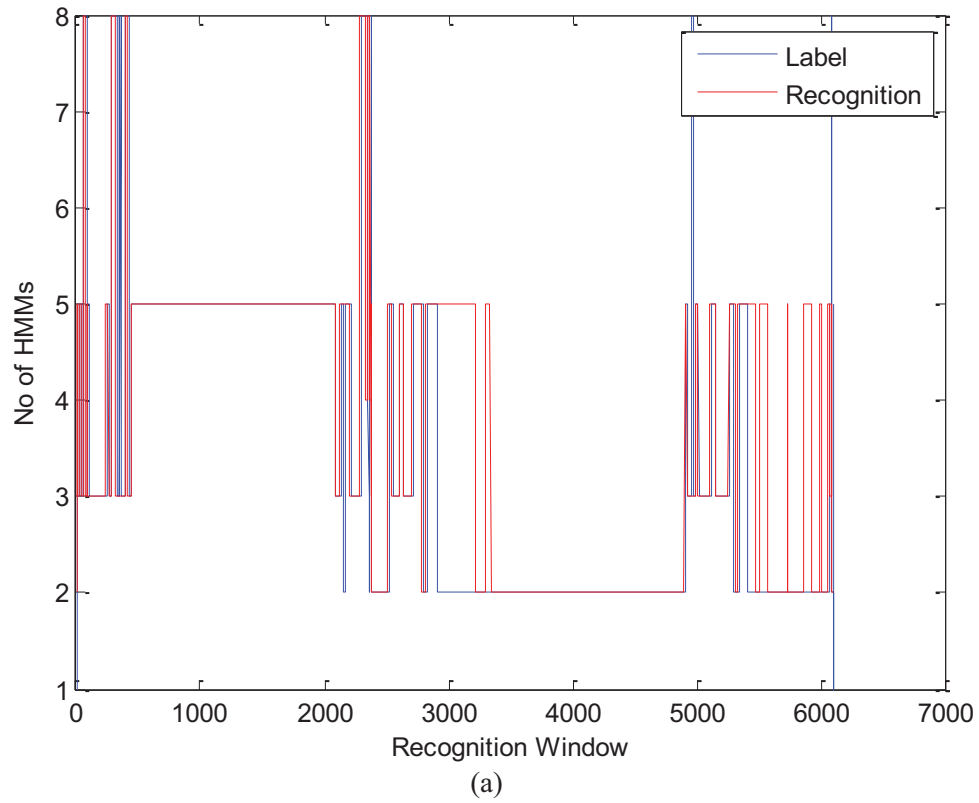
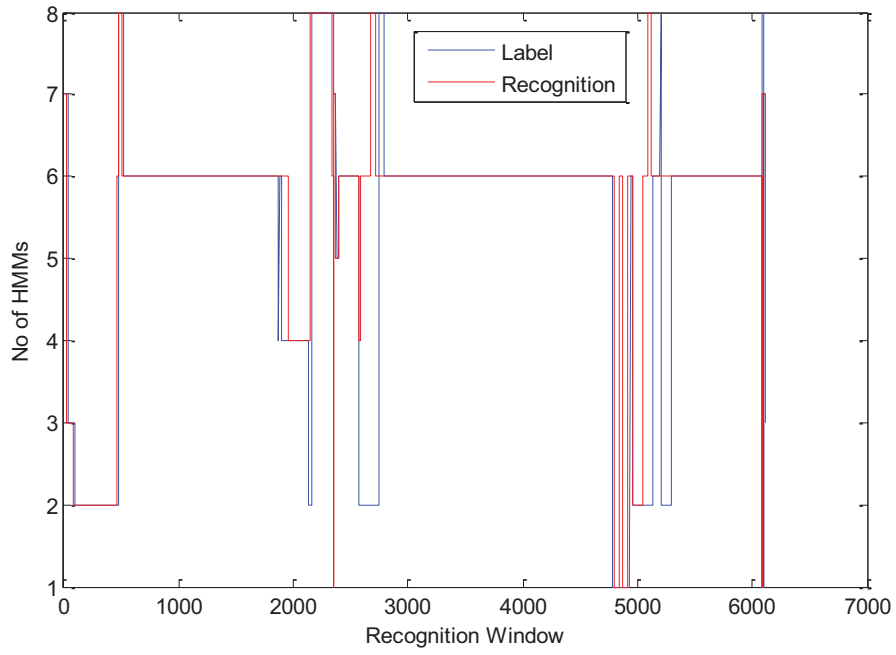
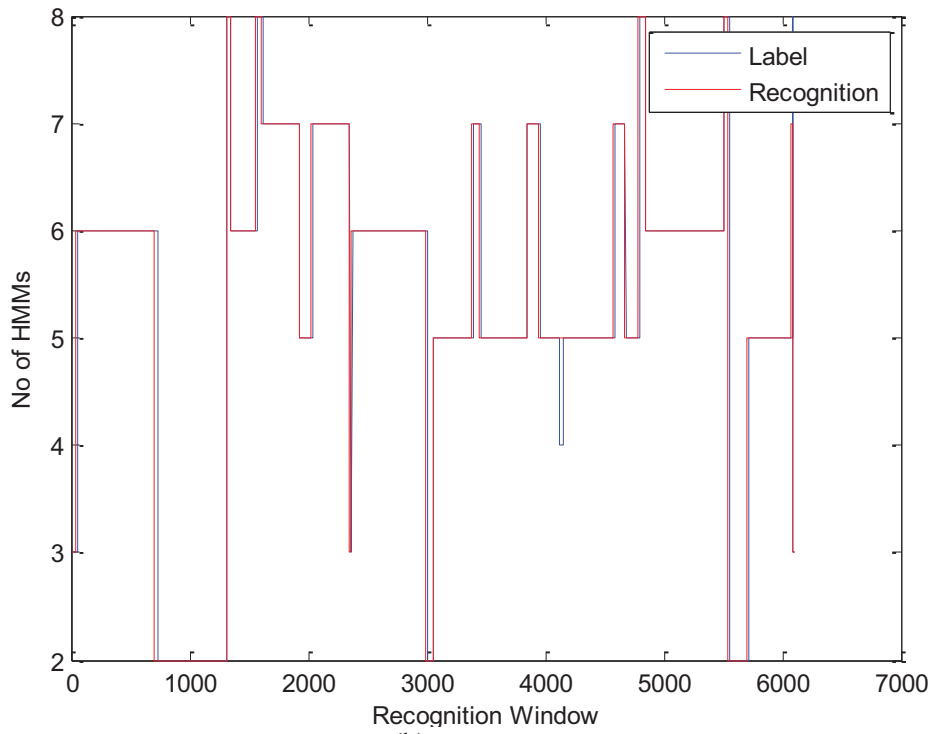


Figure 5-7 Sample of recognized intention for left (a) and right (b) motion trajectory in the frequency domain. Eight HMMs were used to represent the intention. Each HMM was set with 3 Gaussian components and 3 states.



(a)



(b)

Figure 5-8 Sample of recognized intention for left (a) and right (b) motion trajectories in the spatial domain. Eight HMMs were used to represent the intention. Each HMM was configured with 3 states and data were modeled by 3 Gaussian components.

5.3.2.2 Subtask Layer

From above, it is found that the recognition results based on observation feature from the frequency domain is consistent. Recognition rate improves slightly in the spatial domain when the number of states and Gaussian component increases. The training time for recognition model varies with different model configurations. The time can differ by up to an hour with the current experimental data. It is found from the experiment that the training time increases with more states and Gaussian components. Training of the recognition models is often done off line, training time has minor impact to the performance of the overall recognition method. However, in consideration of real time implementation and updating of the recognition model in future, the observation likelihood produced by the primitive recognition model setting with 3 states and 3 Gaussian components was chosen to form observation features for the subtask layer. Figure 5-6 showed that the recognition rate of the primitive recognition model decreases when the number of HMMs increases. This observation has been taken note of and studied its effects to the recognition rate in subtask recognition model.

Emission from Spatial Representation

The subtask recognition model was configured with number of states from 3 to 17, and training data were model with number of Gaussians components from 3 to 9. Figure 5-9 and Figure 5-10 show the recognition results with observation window of 600 frames (14.2 seconds). In Figure 5-9, observation features for the subtask recognition model were formed by the observation probabilities from the primitive recognition model with 8 HMMs. The subtask recognition model with 7 states and 3 Gaussian components is able to produce

the highest average recognition rate of $77.1 \pm 14.5\%$. The recognition rate is low when the recognition model is configured with more than 7 states. This could be because the entire process only needs 6 to 7 states to model it; more states do not help the modelling of the process but made the Expectation Maximization algorithm difficult to determine a global optimal result. In Figure 5-10, the observation features for the subtask recognition model were formed by the observation probabilities from the primitive recognition model with 3 HMMs. It shows that the subtask recognition model with 13 states and 9 Gaussian components is able to produce the highest average recognition rate of $73.7 \pm 12.9\%$.

The effect of observation window width was also studied. Figure 5-11 shows the recognition rate of subtask recognition model with observation window width from 100 to 600 frames. The recognition rate increases by 23.9% and 23.1% for the subtask recognition models based on the 3 and 8 HMMs in primitive recognition model, respectively, as the width of the observation window increases. Figure 5-12 shows an example of results for the subtask recognition.

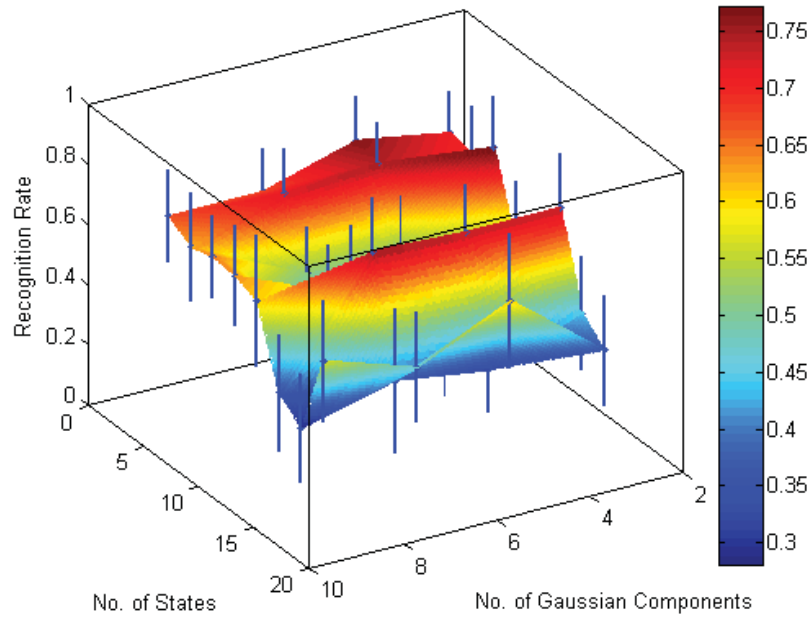


Figure 5-9 Recognition rate of the subtask recognition model trained based on primitive recognition model in the spatial domain. Primitive recognition models were constructed by 8 HMMs. Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.

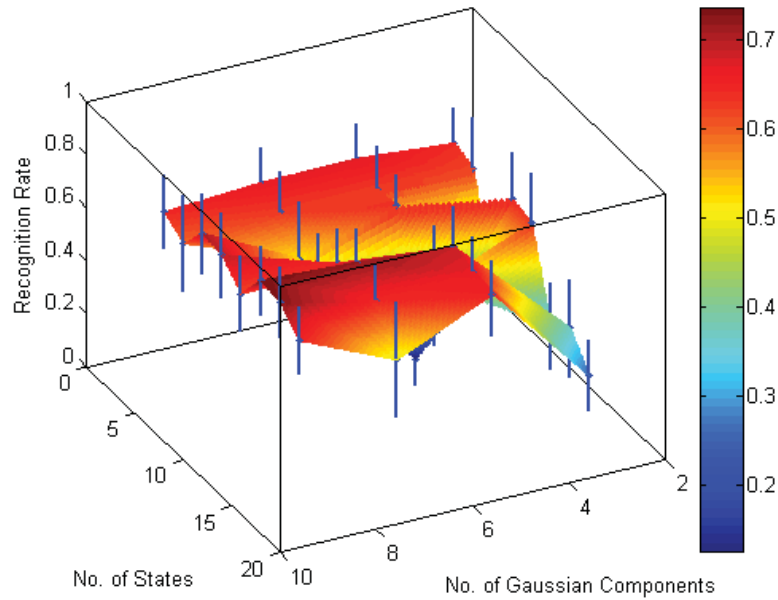


Figure 5-10 Recognition rate of the subtask recognition model trained based on primitive recognition model in the spatial domain. Primitive recognition models were constructed by 3 HMMs, Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.

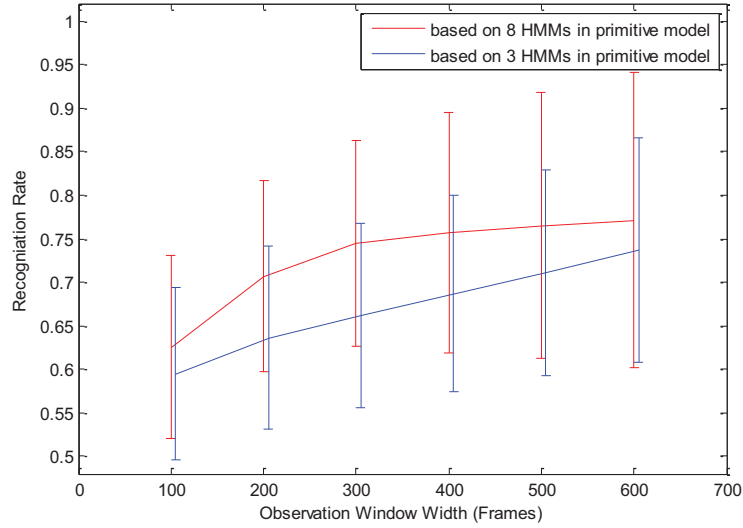


Figure 5-11 Recognition rate with respect to the width of observation window. Red lines: the subtask recognition model was configured with 7 states, three Gaussian components and the respective primitive recognition model was constructed with 8 HMMs. Blue lines: the subtask recognition model was configured with 13 states, nine Gaussian components, and the respective primitive recognition model was constructed by 3 HMMs.

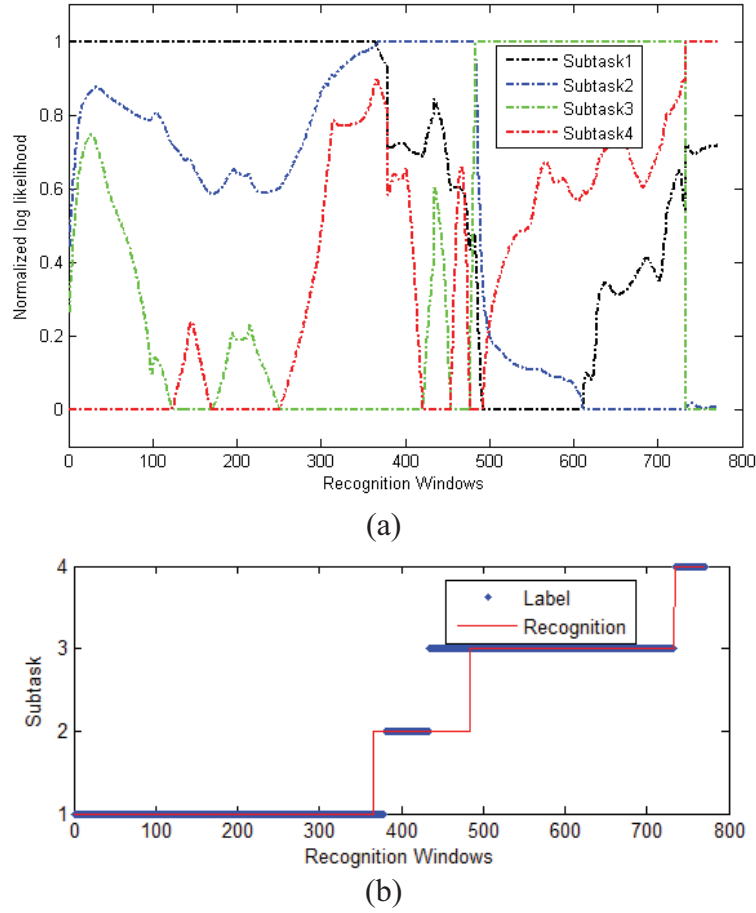


Figure 5-12 Sample of recognized motion intention in subtask level. The subtask recognition model was configured with 7 states and 3 Gaussian components. (a) Normalized log likelihood of four subtasks, (b) recognition result.

Emission from Frequency Representation

Although the primitive recognition model with observation feature in the frequency domain produced high and consistent recognition rate, the recognition rate of the corresponding subtask recognition model is low. Figure 5-13 and Figure 5-14 show the effect of state number and Gaussian components on the recognition rate of the subtask recognition model. Figure 5-13 shows the result based on the primitive recognition model constructed by 3 HMMs. The subtasks recognition model was configured with 7 states and 7 Gaussian components; it produces the highest average recognition rate at only $56.2 \pm 20.0\%$. Figure 5-14 shows the results based on

the primitive recognition model constructed by 8 HMMs. The subtasks recognition model was configured with 9 states and 9 Gaussian components; it produces the highest average recognition rate at only $50.4 \pm 21.1\%$. Figure 5-15 shows the recognition rate of subtask recognition model with observation window width from 100 to 600 frames. The recognition rate increases by 16.8% and 19.1% for the subtask recognition models based on the 3 and 8 HMMs in primitive recognition model, respectively, as the width of the observation window increases.

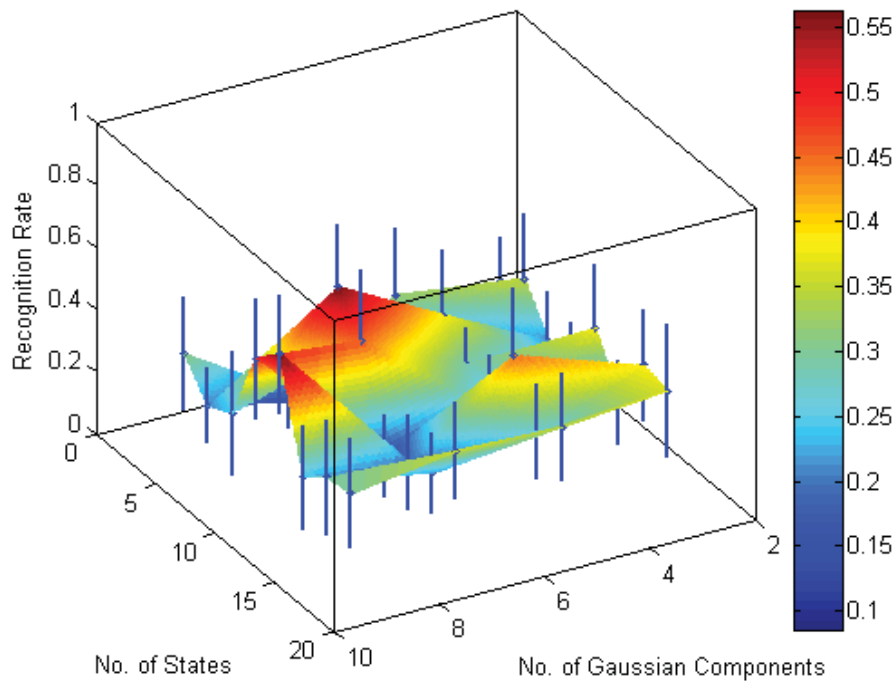


Figure 5-13 Recognition rate of the subtask recognition model trained based on primitive recognition model in the frequency domain. Primitive recognition models were constructed by 3 HMMs. Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.

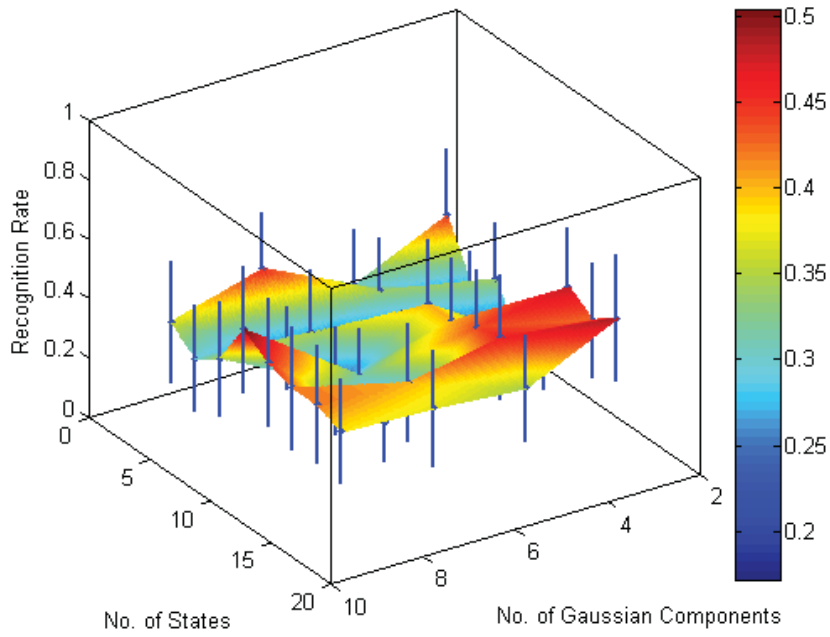


Figure 5-14 Recognition rate of the subtask recognition model trained based on primitive recognition model in the frequency domain. Primitive recognition models were constructed by 8 HMMs. Subtask recognition model was configured with 3 to 17 states and 3 to 9 Gaussian components.

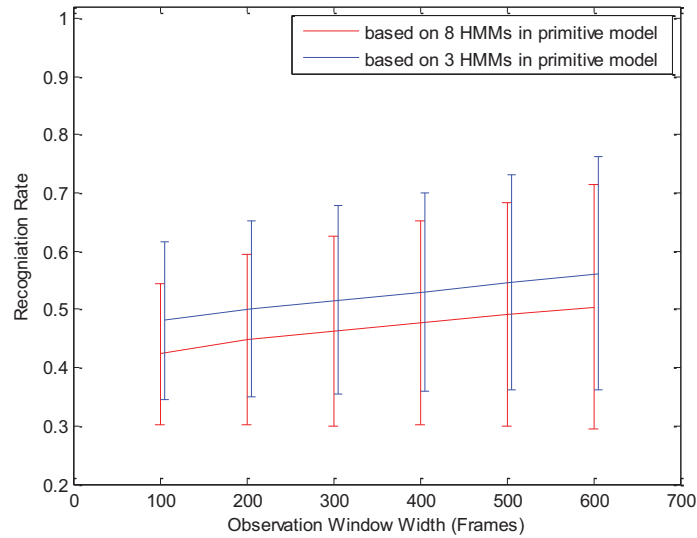


Figure 5-15 Recognition rate with respect to the width of observation window. Red lines: the subtask recognition model was configured with 9 states and 9 Gaussian components and the respective primitive recognition model was constructed with 8 HMMs. Blue lines: the subtask recognition model was configured with 7 states and 7 Gaussian components, and the respective primitive recognition model was constructed by 3 HMMs.

5.4 Summary

A stacked HMM was proposed for motion intention recognition in this chapter. Experiments were conducted on the IRAS training system to study the effectiveness of such recognition model with different configurations and feature representations. In the primitive layer, we observed from Figure 5-6 that the recognition rate decreases as the number of HMMs to construct the recognition model increases for both cases when the features are represented in the frequency domain and spatial domain. Table 5-1 and Table 5-2 show that the number of Gaussian components which represent the observation window data has no significant effect on recognition rate. The number of states in the HMMs in the primitive recognition model also has very limited effect on the recognition rate as shown by Table 5-3 and Table 5-4.

In the subtask recognition layer, the recognition rate is highly affected by the settings of the primitive recognition model. Figure 5-9 and Figure 5-10 show respectively the effect of HMM configuration in subtask recognition model, and effect of HMM numbers used in the primitive recognition model in the spatial domain. Figure 5-13 and Figure 5-14 show such effects in the frequency domain respectively. The performance of the subtask recognition model is better when its primitive recognition model is working in the spatial domain.

Overall, the recognition for primitive recognition model and subtask recognition model are good when the features are represented in the spatial domain. When the features are represented by PSD, the recognition rate for the

primitive is consistently high. The recognition rate for the subtask recognition model is low when the observation likelihood from the primitive recognition model is taken as the observation features. Further investigation on the proposed stacked HMM on IRAS is on-going.

6 SURGICAL SKILLS EVALUATION AND ANALYSIS

In previous chapters, a complete robotic-assisted surgical training system has been discussed. The effectiveness of such a training system is studied and discussed in this chapter. The Image guided Robot-Assisted Surgical (IRAS) training method is effective for surgical training if it could help to enhance the performance of a surgeon. Therefore, the effectiveness of the IRAS training method should be investigated by examining the performance difference between surgeons who have been trained by the IRAS and those who have been not trained by the IRAS training system. An objective assessment method to evaluate the surgeon's performance is required. Manual and computational evaluation methods have been reviewed in Section 2.5. Basic evaluation criteria, such as total task time, path length, path smoothness and traumas on the organs were applied by other researchers in evaluating virtual reality surgical simulators [20] and robotic-assisted surgery [128]. While these evaluation criteria provide information in overall performance, they hardly reflect how close the trainee's performance is to the experienced surgeon's performance, especially from the medical staff's perspective. Another commonly applied technique in evaluating surgical performance is the Hidden Markov Model (HMM) method. Researchers applied HMM technologies in evaluating the proficiency levels of the participants in using LapSim simulator for minimally invasive surgical training [129] and the proficiency levels of surgeons in using the da Vinci surgical system [59].

The objective of this chapter is to study the difference of the IRAS training method and the conventional training method in acquiring laparoscopic

surgical skills. Two experiments were conducted, namely technical evaluation, and clinical evaluation. The technical evaluation aims to validate the hypothesis that the IRAS training method is effective from an engineering perspective. In the technical evaluation, the participants were trained on the IRAS training system and the tests were also performed on the IRAS training system. All participants' surgical procedure was recorded in terms of their motion trajectory. The clinical evaluation aims to prove that the IRAS training method could enhance the medical student's performance in real surgery. Therefore, the participants were trained on the IRAS training system, and they had their tests on real animal model in operating theatre. All participants' surgical procedures were recorded by video and assessed by experienced surgeons.

HMM techniques and Vassiliou's [104] methods were applied to compare the performance of two groups of participants in the technical evaluation and the clinical evaluation, respectively. The rest of this chapter is organized as follows: Section 6.1 discusses the technical evaluation of the training method. Section 6.1.1 - 6.1.3 describes details of technical evaluation including experiment method, HMM evaluation method, and the performance of the participants. Section 6.2 discusses the clinical evaluation. Finally, the work is summarized in Section 6.3.

6.1 Technical Evaluation

HMM was applied to model the surgical skill acquired from an experienced surgeon and evaluated the performance of the participants with the HMM. The

surgical scenario described in Section 5.3.1 was applied for this evaluation experiment.

6.1.1 Evaluation Method

The concept of this evaluation is to use the HMM trained by the experienced surgeon's operation to measure the participants' operation. A virtual surgery conducted by an experienced surgeon was taken to train 4 HMMs λ_{st} for the 4 subtasks described in Section 5.3.1. The trained HMMs λ_{st} were then used to find out the likelihood of the observation sequence $P(O|\lambda_{st})$ extracted from the participants' operation in the same subtask. All the probabilities of the observation sequences were ranked. The ranking indicates the similarity of the participant's operation to the experience surgeon's.

In a manual evaluation, the examiner looks in to how the surgical instruments are positioned relative to the organ and also relative to the other instrument. The HMM evaluation method was designed to evaluate the participants' performance with the features which the examiner are interested in. Hence, observation features for training the HMMs λ_{st} represent the perspective of the examiner. For this study, the observation features include the relative position of the left and right instruments to the specified points on the organ, P_{LO} and P_{RO} ; the position and orientation of the left and right instruments' tip, PO_L and PO_R ; the opened angle of surgical instrument's handle, α_L and α_R ; the angle of the instrument's tip vector to the specific vectors on the organ, β_L and β_R ; the status of footswitch, F_R ; the vector from

left instrument's tip to the right instrument's tip, P_{LR} . These features are illustrated in Figure 6-1. It is expressed as

$$O_{st} = \{P_{LO} \quad P_{RO} \quad PO_L \quad \alpha_L \quad PO_R \quad \alpha_R \quad \beta_L \quad \beta_R \quad P_{LR}\}, \quad (6.1)$$

where st denotes for subtasks in this evaluation process.

These features are selected to capture the operation performed on the organ, such as the places where the instruments operate on, status of the instrument (such as grabbing, ablation, deploying and cutting), relative position of the left instrument with respect to the right instrument and the instruments to organs. It characterizes the tool-tissue relationship and the appropriate way in manipulating the surgical instruments. As a hook electrode is used in subtask 1, the angle of the instrument's handle has no effect to the ablation process. The signal from footswitch and the position of hook electrode can be used to indicate whether the user activates the ablation process at the right places. Therefore the observation feature α_R in expression (6.1) is replaced with the status of footswitch F_R for subtask 1 only.

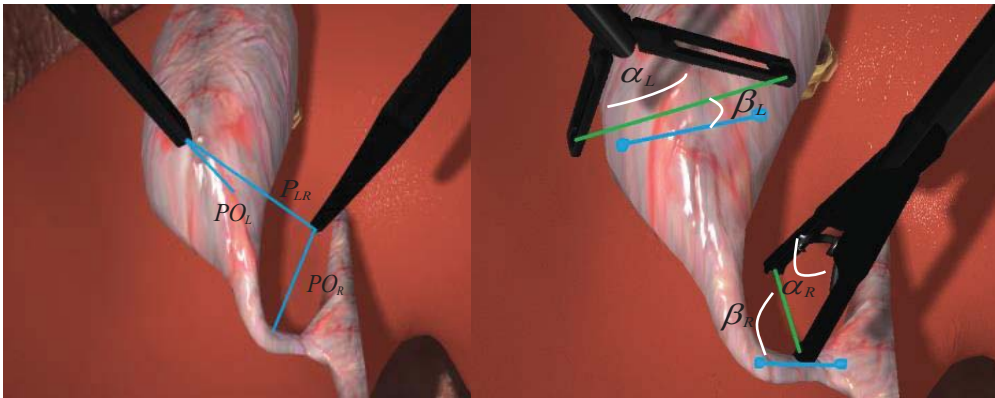


Figure 6-1 Instruments' tip to the specified points on the organ, PO_L and PO_R ; relative position vector from the left instrument's tip to the right instrument's, P_{LR} ; angle between the instrument's tip vector and the specific vectors on the organ, β_L

and β_R ; angle of the instrument handle opened, α_L and α_R , they are proportional to the angle of applicator's jaws formed.

HMM with full transition using Gaussian distribution is applied to model the surgeon's operation procedure. In order to construct HMM models that could adequately represent the surgeon's performance, cross validation is applied to determine the optimal number of states for the HMM models. Each HMM model is set with 3 states as illustrated in Figure 6-2. The initial parameters are estimated by K-means classification method for the Gaussian distribution. The Expectation Maximization algorithm is applied to estimate the parameters of the HMM models [72].

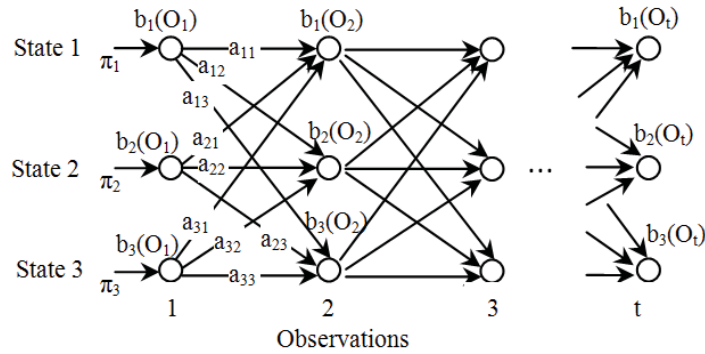


Figure 6-2 Three states full transition HMM. π is the prior probability, a is the state transition matrix and b is the observation probability.

Observation sequences O_{st} extracted from the participants' operation are input into the respective HMM model λ_{st} to find out their probabilities. The probability of the observation sequence generated by the surgeon's HMM model is expressed as $P_{st,i}(O_{st,i} | \lambda_{st})$, where $i=1...120$ is the serial number of the participants' operation. The probability of the observation sequence is taken as a measure of the similarity of the participants' performance to the surgeon's performance.

6.1.2 Experiments

One experienced surgeon and twelve participants participated in the experiment. In the first step of the experiment, the experienced surgeon performed the procedure described in Section 5.3.1 for a total of 10 times. The surgeon performed the virtual surgery with the same requirements as in real surgery such as selecting the tissue grasping location, orientation of the instrument tip during ablation and visibility of the instrument tip for different instruments. The entire virtual surgical procedure was recorded. Features extracted from the acquired data were taken to train HMM models. These HMM models serve as a reference model for comparison with the participants' performances.

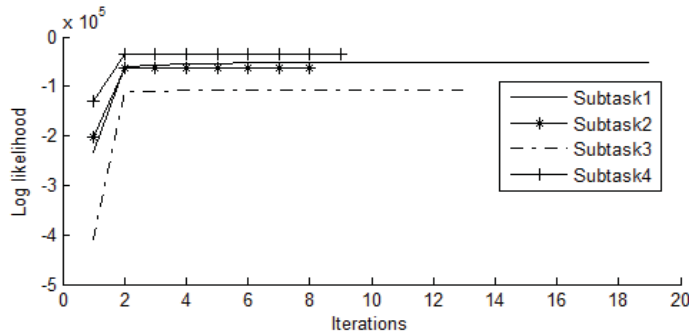


Figure 6-3 Convergence of log likelihood for each subtask in parameter estimation process.

Twelve participants, with an average age of 22.3 ± 3.1 years and no medical experiences participated in the study. They were randomly divided into two groups with 6 participants in each group, namely Group A and Group B. Since neither participant had medical background, they were first introduced with cholecystectomy and the function of each laparoscopic instrument as mentioned in Section 5.3.1. All participants were given 3 hours each to familiarize themselves with the training system one day before the experiment.

In order to obtain a sense of how the laparoscopic instruments worked and how to use the robotic system to perform virtual surgery, they practiced on pointing, grabbing, moving, clipping and dividing operations with the training system. After familiarization with the robotic system, the surgical task described in Section 5.3.1 was then informed to all participants.

Participants in Group A underwent the IRAS training method using the system described in Section 3.1. The handles of the robotic system moved along the recorded trajectories of the experienced surgeon while the virtual surgical scene corresponding to the movement of surgical instruments and tool-organ interaction was also replayed on a wide screen monitor simultaneously. Participants were required to place their hands on the robotic handles to experience the motion of the surgical instrument while watching the surgical simulation scene at the same time. Conversely, the participants in Group B received their training by watching a video of the virtual surgery. The video contains the virtual surgery conducted by the experienced surgeon. All participants were informed to concentrate on how the instruments were manipulated, including relative position to the organ, orientation of the instrument's tip and angle of the instrument's handle opened. The simulated surgical procedures shown to both Group A and B were identical

During the training process, all participants were required to experience one session of training, followed by one session of practice on the virtual patient. This training and practice cycle was repeated 5 times. Upon completion of the 5 cycles of training and practice sessions, all participants were required to

complete 10 tests of the entire task. Therefore, there were 12×5 practices and 120 tests data acquired in total.

6.1.3 Performance Analysis and Discussions

Observation sequences in the form of Equation (6.1) were extracted from the recorded operations of participants. There were 120 observation sequences in total. These observation sequences were fed to the surgeon's HMM model to obtain the probabilities of observation sequences $P_{st,i}(O_{st,i} | \lambda_{st})$, $i = 1 \dots 120$. They were obtained using forward-backward algorithm [72] and expressed in log likelihood. The mean log likelihoods of each of the two groups' test session (from Test Session 1 to Test Session 10) for each subtask are shown in Figure 6-4. The average log likelihoods of the 10 sessions are generally stable. There is no indication showing that the performance gets better (higher log likelihood) when the participants conduct more virtual surgery during the test. Therefore, 5 training sessions are enough for the participants to get familiar with the training system to achieve consistent performance. Participants from Group A generally produced higher log likelihood and smaller standard deviation than that of Group B in all 4 subtasks. These 120 log likelihoods were also ranked from high to low. Percentages of the observation sequences from Group A and their ranking at top 20, top 40 and top 60 of the 120 sets of observation sequences are listed in Table 6-1. Based on the results, Group A obtained the majority of the top 20, 40 and 60 of the 120 sets of observation sequences for all 4 subtasks. This suggests that the participants from Group A tried to execute their virtual surgery in a similar way to the surgeon. The performance of Group A is closer to the surgeon's performance in terms of the observation features.

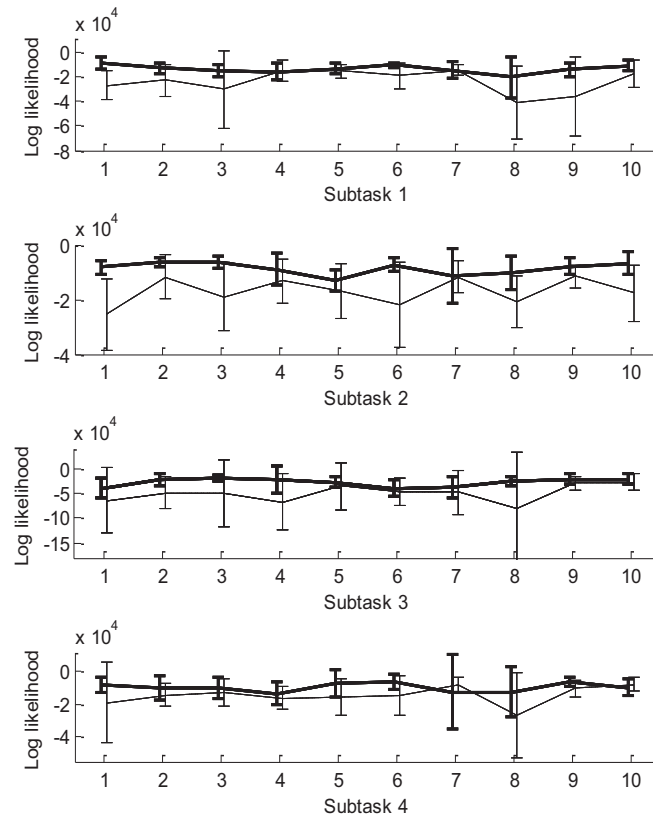


Figure 6-4 Mean log likelihood and standard deviation of each test session. Dark solid lines represent Group A, dash lines represent Group B and vertical bars represent the standard deviation of the likelihood for each test session.

Table 6-1 Percentages of the observation sequences from Group A and ranked at top N of the 120 observation sequences (N=20, 40, 60).

	Top 20	Top 40	Top 60
Subtask 1	65.0%	67.5%	65.0%
Subtask 2	90.0%	77.5%	70.0%
Subtask 3	60.0%	52.5%	53.3%
Subtask 4	70.0%	72.5%	58.3%

Basic evaluation techniques were also applied to measure the performance of the participants. Average task time and trajectory length travelled by the left and right instruments were calculated for both the participants' and the surgeon's operations. They are shown in Table 6-2 and Table 6-3 respectively. The participants in Group B took longer task time and utilized longer trajectory length to complete the task as compared to Group A. This finding

indicates that the participants from Group B may have behaved more hesitantly in performing the tasks; they may have exercised more trial-and-error attempts while navigating the instrument to approach the organ due to the constraints in depth perception [104]. Similarly, due to the limited depth information inherently obtainable from watching training videos, the participants of Group B may have spent more time establishing their sense of depth during the practice and test sessions, resulting in the utilization of longer time and trajectory length. In contrast, Table 6-2 and Table 6-3 show that the task time and trajectory length for participants in Group A to complete the tasks is closer to the surgeon's performance. This might suggest that the motor skills required to perform the tasks have been demonstrated and transferred from the surgeon to the participants.

Using T-test, it is found that the difference between both groups in trajectory length is statistically significant in terms of motion economics ($p < 0.05$). On the other hand, task time may not be an effective evaluation criteria ($p > 0.05$) to identify the difference between both group's performances. Although Table 6-2 shows that there Group A's performance is closer to the surgeon's model than Group B, more tests are required to confirm the results statistically.

Table 6-2 Participants' performance evaluated by average task time, trajectory length of the left and right instruments.

	Participants	Time (s)	Left (mm)	Right (mm)
Test session	Group A	247.6±70.9	579.8±275.4	1578.3±369.0
	Group B	268.4±149.5	978.7±861.4	1850.6±824.0
	p value	0.18	<0.001	0.011

Table 6-3 Surgeon's performance evaluated by average task time, trajectory length of the left and right instruments.

	Time (s)	Left (mm)	Right (mm)
Surgeon	239.5±38	530.1±184.4	1512±144.2

6.2 Clinical Evaluation

Clinical evaluation has been conducted with medical students working on animal models. The aim of the clinical evaluation is to verify the effectiveness of the IRAS training method with real medical scenario. The evaluation method described in [104] was applied in this evaluation.

6.2.1 Experimental Method

Twelve medical students participated in the clinical evaluation. They were not trained with practical laparoscopic skills prior to this study. All participants were randomly divided into 2 groups, namely the Control Group and the IRAS Group. The two groups of participants underwent the experimental procedure as described in Table 6-4. Each participant in both groups watched a video taken in a real laparoscopic surgery for 1 hour before the exact date of experiment. The laparoscopic procedure in the video was conducted by an expert laparoscopic surgeon. The major difference of the experimental procedure focused on Step 2 where the Control Group received training with laparoscopic train box (as shown in Figure 6-5) which are commonly used in the medical curriculum, and the IRAS Group received training with the IRAS training system. Each participant was given two hours in Step 2. The Control Group spent two hour on the laparoscopic training box with fundamental laparoscopic skill. The IRAS Group spent their first half

hour in getting adapted to the operation of the IRAS training system and the simulation environment. In the rest of the one and a half hour, the participant was first guided by a pre-recorded procedure to complete a surgical procedure (The pre-recorded procedure was performed by an expert surgeon prior to this training session). After the guided training, the participant was required to practice on his / her own using the IRAS training system. This procedure was repeated in the remaining 1.5 hours. Once the training session was completed, all participants were required to perform a laparoscopic cholecystectomy on a real porcine model as stated in Table 6-4 step 3. The entire surgical procedures carried out by these two groups of participants in Step 3 were recorded by video camera for manual evaluation.



Figure 6-5 Laparoscopic training boxes for Control Group's training.

Table 6-4 Summary of experiment procedure for clinical evaluation.

	Control Group	IRAS Group
Step1	Each participant watched a video (1) of laparoscopic cholecystectomy for 1 hour one day prior the training	Each participant watched a video (1) of laparoscopic cholecystectomy for 1 hour one day prior the training.
Step2	Each participant received training and practiced on laparoscopic train box for 2 hours.	Each participant received the IRAS training for 2 hours (First 0.5 hour: get adapt to the operation of the simulation system. The following 1.5 hours: training and practice).
Step3	Each participant performed a laparoscopic cholecystectomy on a porcine model.	Each participant performed a laparoscopic cholecystectomy on a porcine model.
Note	(1): the video contains a full procedure of a laparoscopic cholecystectomy conducted by an expert surgeon.	

6.2.2 Performance Analysis and Discussions

The procedural videos recorded in Table 6-4 step 3 were evaluated according to the assessment criteria shown in Table 2-1 and Table 2-2 [104]. In order to minimize any possible subjective bias during evaluation, the evaluators who assessed the procedural videos were not involved in the project development, and they were not given any information on the grouping information of the participants either. Table 6-5 shows the average score of the two groups over 5 global assessment criteria. Besides the 5 global assessment criteria, the entire procedure was divided into 10 subtasks as stated in Table 2-2, each subtask was given a mark of 1 or 0 which corresponding to done or not done. Table 6-6 shows the average score of the two groups. Note that the procedural videos of participant no. 6 in the Control Group and participant no. 4 in the IRAS Group were corrupted and hence, they were excluded from the study.

The evaluation results in Table 6-5 and Table 6-6 show that the IRAS Group has received higher marks than that of the Control Group in both the global assess scale and the subtask assessment scale. This experimental result suggests that the IRAS Group generally performed better than the Control Group. Hence, we can conclude that the IRAS training method is effective.

Table 6-5 Average score of the students in the surgeries.

	Reviewer1		Reviewer2		Average	
	Control group	IRAS group	Control group	IRAS group	Control group	IRAS group
Depth Perception /5	2.8±0.84	3.2±0.84	3.2±0.84	3.6±0.89	3.0±0.82	3.4±0.84
Bimanual dexterity /5	2.0±0.71	2.2±0.84	3.2±0.84	3.6±0.55	2.6±0.97	2.9±0.99
Efficiency /5	2.8±0.45	3.2±0.84	3.8±0.45	3.4±1.52	2.9±0.68	3.3±1.16
Tissue Handling /5	2.0±0.71	2.8±0.45	3.2±0.84	3.4±1.14	2.6±0.97	3.1±0.88
Autonomy /5	3.0±0.00	3.0±0.00	4.8±0.45	4.2±0.84	3.9±0.99	3.6±0.84
Total (25 marks)	12.6±1.52	14.4±2.61	18.2±2.77	18.2±4.60	15.4±3.63	16.3±4.06

Note: each assessment criteria carries 5 marks, with 5 represents the best.

Table 6-6 Average score of 10 subtasks.

	Reviewer1		Reviewer2		Average	
	Control group	IRAS group	Control group	IRAS group	Control group	IRAS group
Subtask1	0.6±0.55	0.8±0.45	1.0±0.00	1.0±0.00	0.8±0.42	0.9±0.32
Subtask2	0.8±0.45	1.0±0.00	0.8±0.45	1.0±0.00	0.8±0.42	1.0±0.00
Subtask3	0.6±0.55	0.6±0.55	1.0±0.00	1.0±0.00	0.8±0.42	0.8±0.42
Subtask4	0.6±0.55	0.6±0.55	1.0±0.00	1.0±0.00	0.8±0.42	0.8±0.42
Subtask5	0.0±0.00	0.2±0.45	0.4±0.55	0.8±0.45	0.2±0.42	0.5±0.53
Subtask6	0.0±0.00	0.0±0.00	0.0±0.00	0.2±0.45	0.0±0.00	0.1±0.31
Subtask7	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00
Subtask8	0.6±0.55	0.6±0.55	1.0±0.00	0.6±0.55	0.8±0.42	0.6±0.52
Subtask9	0.2±0.45	1.0±0.00	0.8±0.45	1.0±0.00	0.5±0.52	1.0±0.00
Subtask10	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00	1.0±0.00
Total (10 marks)	5.4±1.52	6.8±0.84	8.0±1.22	8.6±0.89	6.7±1.89	7.7±1.25

Note: see Table 2-2 for the details of the subtasks.

There are several limitations raised in the experiment: (1) Levels of operation difficulty to each participant were different. The level of operation difficulty is subjected to the morphology of the porcine model. One of the solutions to mitigate the effect of such differences is to normalize the

evaluation score by a difficulty level. However, the difficulty level assigned to the task is still very subjective; and (2) the number of experiment participants and evaluator is small. A larger scale of such experiment is required to prove that the experimental results are statistically significant. Although the experiment suggested that the IRAS training method is effective, further experiments are required due to the above limitations.

6.3 Summary

Experiments were conducted on the IRAS training system to compare the efficiency between the IRAS training method and conventional training method.

In technical evaluation, participants who received the IRAS training took lesser task time and shorter trajectory length to complete the tasks. The difference in utilized trajectory length between the two groups has been found to be statistically significant. HMM was also applied to characterize and compare the performance of participants with the surgeon's. Group A that received the IRAS training produced higher average probability of observation sequence as compared to Group B. The results suggest that the IRAS training system is more effective in transferring motor skills to the user than that of the conventional training method. In this technical evaluation, relatively simple tasks were used for the evaluation. More comprehensive scenarios can be constructed for future studies.

A well accepted clinical evaluation method was applied in the clinical evaluation. The clinical evaluation also shows that the performance of the participants trained by the IRAS training method is better than that of the

participants trained by conventional methods. However, the number of participants and reviewer in the clinical evaluation is limited. More participants and reviewer's participation are required to draw a conclusion with statistical significance.

7 DISCUSSION AND CONCLUSION

This thesis systematically proposed, designed, developed and validated an innovative surgical training method (IRAS) described in Chapter 1. Customized surgical simulation robot and surgical simulation system were built for the IRAS training method. The human operator and robot collaborated in this training method. Both technical evaluation and clinical evaluation have shown that the IRAS training method is effective in transferring the motor skills from expert surgeon to novice surgeons. However, there are limitations with the IRAS training system.

The robotic surgical trainer was built with the capability of recording and rendering the recorded instruments' trajectory precisely, and the robot was also built with the capability of haptic output to render the tool tissue interaction for the human operator to obtain a sense of interacting with real objects. New motion learning and intention recognition algorithms were proposed and investigated. However, the robotic surgical trainer still lacks the capabilities to conduct comprehensive training, such as assisted training and assessment of performance with experts' knowledge automatically. The robot needs to be equipped with more intelligence to understand and react with the human operator for specific surgical procedure. This may involve the development of a specific cognitive engine that possesses the situational knowhow of an expert surgeon.

The IRAS training method has been validated through technical evaluation and clinical evaluation. Both studies show that the participants' performance is better while trained by the IRAS training method. The participants trained by

the IRAS required shorter time and tool travelling length in completion of same tasks, and they also received higher marks when they were assessed by the expert surgeon with the clinical assessment criteria. However, the participants trained by the IRAS obtained slightly lower marks when assessed by *Autonomy* [104] in the clinical evaluation. There could be risks that the trainee may develop dependency to the robotic guided motion. This risk needs to be taken care of when deploying the system for medical education. Training curriculum shall be carefully designed so that the IRAS training method can help the trainee shorten the learning curve and still retain his / her autonomy in a surgery.

The technical evaluation compared the similarity of the participants' performance with the surgeon's performance in terms of observation features designed. This evaluation method could not identify the critical steps since all steps are assigned equal weight. Stringent requirement shall be imposed for surgical procedure. When the observations from different participants produce very closed log likelihoods, the method used in the technical evaluation may not be able to identify the participants who did not perform well on the critical steps. Assessment method used in the clinical evaluation gives an overall assessment of the surgical procedure from different aspects. The marks given by the evaluator also makes this method subjective to the bias of the evaluator.

Due to the limitation of resources, clinical evaluation was only conducted with 6 participants in each group. Each participant operated on one porcine model. Each porcine model presented unique features for the surgery. The difficulty level of each surgery is therefore different. It imposed a larger variation on the testing environment to our clinical evaluation. Although the

expert surgeons who examine the performance of the trainee have already tried to take this into consideration, the effects of such variations are hard to be completely eliminated. Furthermore, this consideration is also subjected to the opinion of individual evaluator. Therefore, a large scale multiple institutions study with more participants and examiners is preferred to eliminate the effects of such variation. The clinical evaluation is very time consuming, labor and cost intensive.

A fully intelligent system with more training scenarios and objective assessment method will contribute to the provision of consistently high quality surgical training, and relieve the workload of medical staff for medical education. Our robotic surgical trainer could serve as a general platform for the laparoscopic surgical training and simulation. It can be used for other laparoscopic simulation scenarios in addition to cholecystectomy. Medical images processing and real time surgical simulation will be required to build more surgical scenarios.

REFERENCES

1. J. Vozenilek, J. S. Huff, M. Reznick, and J. A. Gordon, "See one, do one, teach one: advanced technology in medical education," *Acad Emerg Med*, vol. 11, pp. 1149-54, Nov 2004.
2. D. G. Murphy, R. Hall, R. Tong, R. Goel, and A. J. Costello, "Robotic technology in surgery: current status in 2008," *ANZ J Surg*, vol. 78, pp. 1076-81, Dec 2008.
3. J. Winer, M. F. Can, D. L. Bartlett, H. J. Zeh, and A. H. Zureikat, "The current state of robotic-assisted pancreatic surgery," *Nat Rev Gastroenterol Hepatol*, Jun 26 2012.
4. J. Liu, S. C. Cramer, and D. J. Reinkensmeyer, "Learning to perform a new movement with robotic assistance: comparison of haptic guidance and visual demonstration," *J Neuroeng Rehabil*, vol. 3, p. 20, 2006.
5. E. O. Kwon, T. C. Bautista, H. Jung, R. Z. Goharderakhshan, S. G. Williams, and G. W. Chien, "Impact of robotic training on surgical and pathologic outcomes during robot-assisted laparoscopic radical prostatectomy," *Urology*, vol. 76, pp. 363-8, Aug.2010
6. C. L. Teo, E. Burdet, and H. P. Lim, "A robotic teacher of Chinese handwriting," in *proceedings of the 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, 2002, pp. 335-341.
7. J. Bluteau, S. Coquillart, Y. Payan, and E. Gentaz, "Haptic guidance improves the visuo-manual tracking of trajectories," *PLoS One*, vol. 3, p. e1775, 2008.
8. C. R. Voyles, A. B. Petro, A. L. Meena, A. J. Haick, and A. M. Koury, "A practical approach to laparoscopic cholecystectomy," *Am J Surg*, vol. 161, pp. 365-70, Mar 1991.
9. L. Bencini, M. Bernini, F. Martini, M. Rossi, C. Tommasi, E. Miranda, L. J. Sanchez, R. Naspetti, R. Manetti, A. Ferrara, S. Nesi, B. Boffi, M. Farsi, and R. Moretti, "Laparoscopic appendectomy performed by residents and experienced surgeons," *JSLs*, vol. 13, pp. 391-7, Jul-Sep 2009.
10. A. M. Derossis, G. M. Fried, M. Abrahamowicz, H. H. Sigman, J. S. Barkun, and J. L. Meakins, "Development of a model for training and evaluation of laparoscopic skills," *Am J Surg*, vol. 175, pp. 482-7, Jun 1998
11. CAE, 2013. <http://www.caehealthcare.com/eng/surgical-simulators/lapvr>.
12. Simbionix, 2014. <http://simbionix.com/simulators/lap-mentor/>.

13. A. Baheti, S. Seshadri, A. Kumar, G. Srimathveeravalli, T. Kesavadas, and K. Guru, "RoSS: Virtual Reality Robotic Surgical Simulator for the da Vinci Surgical System," in symposium on Haptic interfaces for virtual environment and teleoperator systems, 2008, pp. 479-480.
14. C. S. Lee, L. Yang, T. Yang, C. K. Chui, J. Liu, W. Huang, Y. Su, and S. K. Chang, "Designing an active motor skill learning platform with a robot-assisted laparoscopic trainer," Conf Proc IEEE Eng Med Biol Soc, vol. 2011, pp. 4534-7, 2011.
15. Bharath Chakravarthy, Elizabeth ter Haar, Srinidhi Subraya Bhat, Christopher Eric McCoy, and S. L. T. Kent Denmark, "Simulation in Medical School Education: Review for Emergency Medicine," Western Journal of Emergency Medicine, vol. 12, p. 5, Nov 2011.
16. J. L. Lane, S. Slavin, and A. Ziv, "Simulation in Medical Education: A Review," Simulation & Gaming, vol. 32, pp. 297-314, September 1, 2001.
17. Y. Okuda, E. O. Bryson, S. DeMaria, Jr., L. Jacobson, J. Quinones, B. Shen, and A. I. Levine, "The utility of simulation in medical education: what is the evidence?," Mt Sinai J Med, vol. 76, pp. 330-43, Aug 2009.
18. Seiichi Ikeda, Carlos Tercero Villagran, Toshio Fukuda, Yuta Okada, Fumihito Arai, Makoto Negoro, Motoharu Hayakawa, and I. Takahashi, "Patient-Specific IVR Endovascular Simulator with Augmented Reality for Medical Training and Robot Evaluation " Journal of Robotics and Mechatronics, vol. 20, p. 8, 2008.
19. W. I. Willaert, R. Aggarwal, I. Van Herzeele, N. J. Cheshire, and F. E. Vermassen, "Recent advancements in medical simulation: patient-specific virtual reality simulation," World J Surg, vol. 36, pp. 1703-12, Jul 2012.
20. L. M. Sutherland, P. F. Middleton, A. Anthony, J. Hamdorf, P. Cregan, D. Scott, and G. J. Maddern, "Surgical simulation: a systematic review," Ann Surg, vol. 243, pp. 291-300, Mar 2006.
21. <http://thecgroup.com/product/general-gastrointestinal/robotic-2/robotic-trainer>.
22. C. S. Lee, L. Yang, T. Yang, C. K. Chui, J. Liu, W. Huang, Y. Su, and S. K. Chang, "Designing an active motor skill learning platform with a robot-assisted laparoscopic trainer," Conf Proc IEEE Eng Med Biol Soc, vol. 2011, pp. 4534-7, 2011.
23. C. Chee-Kong, C. Chin-Boon, Y. Tao, W. Rong, H. Weimin, J. Liu, S. Yi, and S. Chang, "Learning laparoscopic surgery by imitation using

- robot trainer," in IEEE International Conference on Robotics and Biomimetics (ROBIO), 2011, pp. 2981-2986.
24. Y. Okada, S. Ikeda, T. Fukuda, F. Arai, M. Negoro, and I. Takahashi, "Photoelastic Stress Analysis on Patient-Specific Anatomical Model of Cerebral Artery," in International Symposium on Micro-NanoMechatronics and Human Science, MHS '07, 2007 pp. 538-543.
 25. S. Schendel, K. Montgomery, A. Sorokin, and G. Lionetti, "A surgical simulator for planning and performing repair of cleft lips," J Craniomaxillofac Surg, vol. 33, pp. 223-8, Aug 2005.
 26. J. Pettersson, K. L. Palmerius, H. Knutsson, O. Wahlstrom, B. Tillander, and M. Borga, "Simulation of patient specific cervical hip fracture surgery with a volume haptic interface," IEEE Trans Biomed Eng, vol. 55, pp. 1255-65, Apr 2008.
 27. S. Suzuki, K. Eto, A. Hattori, K. Yanaga, and N. Suzuki, "Surgery simulation using patient-specific models for laparoscopic colectomy," Studies in Health Technology and Informatics, vol. 125, pp. 464-466, 2007.
 28. F. Auricchio, M. Conti, M. De Beule, G. De Santis, and B. Verhegghe, "Carotid artery stenting simulation: from patient-specific images to finite element analysis," Med Eng Phys, vol. 33, pp. 281-9, Apr 2010.
 29. J. H. Palep, "Robotic assisted minimally invasive surgery," J Minim Access Surg, vol. 5, pp. 1-7, Jan 2009.
 30. P. S. J. Louis, L. S. Chiang, T. Shouwei, H. K. Yu, and C. S. e, "Surgical Robotic System for Flexible Endoscopy ", patent, PCT/SG2007/000081, WO2007111571, Ed., 2007, p. 37.
 31. K. Hyosig and J. T. Wen, "Robotic assistants aid surgeons during minimally invasive procedures," IEEE Engineering in Medicine and Biology Magazine, vol. 20, pp. 94-104, 2001.
 32. X. Liu, M. Balicki, R. H. Taylor, and J. U. Kang, "Automatic online spectral calibration of Fourier-domain OCT for robotic surgery," Proceedings of SPIE, 2011, vol. 7890, p. 875210, 2011.
 33. M. Hermann, G. Faustino, W. Daan, N. Istvan, K. Alois, and S. Jurgen, "A System for Robotic Heart Surgery that Learns to Tie Knots Using Recurrent Neural Networks," in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 543-548, 2006.
 34. da Vinci Surgical System <http://www.intuitivesurgical.com/products/>.
 35. M. C. Porte, G. Xeroulis, R. K. Reznick, and A. Dubrowski, "Verbal feedback from an expert is more effective than self-accessed feedback

about motion efficiency in learning new surgical skills," *Am J Surg*, vol. 193, pp. 105-10, Jan 2007.

36. A. M. Derossis, J. Bothwell, H. H. Sigman, and G. M. Fried, "The effect of practice on performance in a laparoscopic simulator," *Surg Endosc*, vol. 12, pp. 1117-20, Sep 1998.
37. D. T. Woodrum, P. B. Andreatta, R. K. Yellamanchilli, L. Feryus, P. G. Gauger, and R. M. Minter, "Construct validity of the LapSim laparoscopic surgical simulator," *The American Journal of Surgery*, vol. 191, pp. 28-32, 2006.
38. A. Liu, Y. Bhasin, and M. Bowyer, "A haptic-enabled simulator for cricothyroidotomy," *Stud Health Technol Inform*, vol. 111, pp. 308-13, 2005.
39. R. Lapeer, M. S. Chen, and J. Villagrana, "An augmented reality based simulation of obstetric forceps delivery," in *Third IEEE and ACM International Symposium on Mixed and Augmented Reality, ISMAR 2004*, pp. 274-275.
40. A. Liu, F. Tendick, K. Cleary, and C. Kaufmann, "A Survey of Surgical Simulation: Applications, Technology, and Education," *Presence: Teleoperators and Virtual Environments*, vol. 12, pp. 599-614, Dec 2003.
41. J. Solis, C. A. Avizzano, and M. Bergamasco, "Teaching to write Japanese characters using a haptic interface," in *proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, HAPTICS 2002*, pp. 255-262.
42. D. Wang, Y. Zhang, and C. Yao, "Machine-mediated Motor Skill Training Method in Haptic-enabled Chinese Handwriting Simulation System," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5644-5649, 2006.
43. A. Basteris, L. Bracco, and V. Sanguineti, "Robot-assisted intermanual transfer of handwriting skills," *Hum Mov Sci*, vol. 31, pp. 1175-90, 2012.
44. R. Sigrist, G. Rauter, R. Riener, and P. Wolf, "Augmented visual, auditory, haptic, and multimodal feedback in motor learning: A review," *Psychon Bull Rev*, vol. 20, pp. 21-53, Nov 7 2012.
45. C. Basdogan, S. De, J. Kim, M. Manivannan, H. Kim, and M. A. Srinivasan, "Haptics in minimally invasive surgical simulation and training," *IEEE Computer Graphics and Applications*, vol. 24, pp. 56-64, 2004.
46. M. P. Schijven and J. J. Jakimowicz, "Introducing the Xitact LS500 laparoscopy simulator: toward a revolution in surgical education," *Surg Technol Int*, vol. 11, pp. 32-6, 2003.

47. Lap Mentor, 2014. <http://simbionix.com/simulators/lap-mentor/>.
48. L. J. M. v. d. Bedem, Hendrix, R., Naus, G.J.L., Aalst, R. van der, Rosielle, P.C.J.N., Nijmeijer, H., Maessen, J.G., Broeders, I.A.M.J. & Steinbuch, M., "Sofie, a robotic system for minimally invasive surgery," in the 6th International MIRA Congress, Athens, Greece, 2011 p. 056.
49. da Vinci Surgical Robot, <http://www.davincisurgery.com/>.
50. K. Hyosig and J. T. Wen, "EndoBot: a robotic assistant in minimally invasive surgeries," in proceedings IEEE International Conference on Robotics and Automation, 2011, pp. 2031-2036 vol.2.
51. S. J. Phee, K. Y. Ho, D. Lomanto, S. C. Low, V. A. Huynh, A. P. Kencana, K. Yang, Z. L. Sun, and S. C. Chung, "Natural orifice transgastric endoscopic wedge hepatic resection in an experimental model using an intuitively controlled master and slave transluminal endoscopic robot (MASTER)," Surg Endosc, vol. 24, pp. 2293-8, 2009.
52. S. Schaal, "Is imitation learning the route to humanoid robots?," Trends in Cognitive Sciences, vol. 3, pp. 233-242, 1999.
53. B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," Robotics and Autonomous Systems, vol. 57, pp. 469-483, May 31 2009.
54. M. Hermann, G. Faustino, W. Daan, N. Istvan, K. Alois, and S. Jurgen, "A System for Robotic Heart Surgery that Learns to Tie Knots Using Recurrent Neural Networks," in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 543-548, 2006.
55. J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," Neural Netw, vol. 21, pp. 682-97, May 2008.
56. J. Kober and J. Peters, "Imitation and Reinforcement Learning," IEEE Robotics & Automation Magazine, vol. 17, pp. 55-62, 2010.
57. E. Theodorou, J. Buchli, and S. Schaal, "Reinforcement learning of motor skills in high dimensions: A path integral approach," in 2010 IEEE International Conference on Robotics and Automation (ICRA), 2010, pp. 2397-2403.
58. P. Kormushev, S. Calinon, and D. G. Caldwell, "Robot motor skill coordination with EM-based Reinforcement Learning," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2010 pp. 3232-3237.
59. C. E. Reiley, E. Plaku, and G. D. Hager, "Motion generation of robotic surgical tasks: Learning from expert demonstrations," in 2010 Annual

International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2010, pp. 967-970.

60. T. Inamura, N. Kojo, T. Sonoda, K. Sakamoto, K. Okada, and M. Inaba, "Intent imitation using wearable motion capturing system with on-line teaching of task attention," in 5th IEEE-RAS International Conference on Humanoid Robots, 2005, pp. 469-474.
61. S. Calinon, F. Guenter, and A. Billard, "On Learning, Representing, and Generalizing a Task in a Humanoid Robot," IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics, vol. 37, pp. 286-298, 2007.
62. K. Yamane, Y. Yamaguchi, and Y. Nakamura, "Human motion database with a binary tree and node transition graphs," Autonomous Robots, vol. 30, pp. 87-98, 2011.
63. C. E. Reiley, H. C. Lin, B. Varadarajan, B. Vagvolgyi, S. Khudanpur, D. D. Yuh, and G. D. Hager, "Automatic recognition of surgical motions using statistical modeling for capturing variability," Stud Health Technol Inform, vol. 132, pp. 396-401, 2008.
64. H. C. Lin, I. Shafran, D. Yuh, and G. D. Hager, "Towards automatic skill evaluation: detection and segmentation of robot-assisted surgical motions," Comput Aided Surg, vol. 11, pp. 220-30, Sep 2006.
65. S. Calinon, F. D'Halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and Reproduction of Gestures by Imitation," IEEE Robotics & Automation Magazine, vol. 17, pp. 44-54, June 2010.
66. S. Calinon and A. Billard, "Stochastic gesture production and recognition model for a humanoid robot," in proceedings 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004, pp. 2769-2774 vol.3.
67. H. Hajimirsadeghi, M. N. Ahmadabadi, M. Ajallooeian, B. N. Araabi, and H. Moradi, "Conceptual Imitation Learning: An Application to Human-Robot Interaction," in 2nd Asian Conference on Machine Learning, Tokyo, Japan, 2010, pp. 331-346.
68. C. S. Hundtofte, G. D. Hager, and A. M. Okamura, "Building a task language for segmentation and recognition of user input to cooperative manipulation systems," in proceedings 10th Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2002, pp. 225-230.
69. D. Aarno, S. Ekvall, and D. Kragic, "Adaptive Virtual Fixtures for Machine-Assisted Teleoperation Tasks," in proceedings of the IEEE International Conference on Robotics and Automation, 2005, pp. 1139-1144.

70. S. Calinon and A. Billard, "Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA, and HMM," in the International Conference on Machine Learning, 2005, pp. 105-112.
71. L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, pp. 257-286, 1989.
72. L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Magazine*, vol. 3, pp. 4-16, 1986.
73. T. Y. Chen, X. D. Mei, J. S. Pan, and S. H. Sun, "Optimization of HMM by the tabu search algorithm," *Journal of Information Science and Engineering*, vol. 20, pp. 949-957, Sep 2004.
74. N. Thatphithakkul and S. Kanokphara, "HMM parameter optimization using tabu search [speech recognition]," in *IEEE International Symposium on Communications and Information Technology*, 2004, pp. 904-908 vol.2.
75. Y. Fengqin, Z. Changhai, and B. Ge, "A Novel Genetic Algorithm Based on Tabu Search for HMM Optimization," in *Fourth International Conference on Natural Computation*, 2008, pp. 57-61.
76. Y. Fengqin and Z. Changhai, "An Effective Hybrid Optimization Algorithm for HMM," in *Fourth International Conference on Natural Computation*, 2008, pp. 80-84.
77. S. Calinon, F. D'Halluin, D. G. Caldwell, and A. G. Billard, "Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework," in *9th IEEE-RAS International Conference on Humanoid Robots*, 2009, pp. 582-588.
78. A. G. Billard, S. Calinon, and F. Guenter, "Discriminative and adaptive imitation in uni-manual and bi-manual tasks," *Robotics and Autonomous Systems*, vol. 54, pp. 370-384, 2006.
79. F. Sadri, "Logic-Based Approaches to Intention Recognition," in *Handbook of Research on Ambient Intelligence and Smart Environments: Trends and Perspectives*: IGI Global, pp. 346-375, 2011.
80. D. Aarno and D. Kragic, "Motion intention recognition in robot assisted applications," *Robotics and Autonomous Systems*, vol. 56, pp. 692-705, 2008.
81. R. Kelley, M. Nicolescu, A. Tavakkoli, C. King, and G. Bebis, "Understanding human intentions via Hidden Markov Models in autonomous mobile robots," in the *3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2008, pp. 367-374.

82. O. C. Schrempf, D. Albrecht, and U. D. Hanebeck, "Tractable probabilistic models for intention recognition based on expert knowledge," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2007, pp. 1429-1434.
83. M. Wais and D. Henrich, "Human-robot collaboration by intention recognition using probabilistic state machines," in IEEE 19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD), 2010, pp. 75-80.
84. K. Tahboub, "Intelligent Human-Machine Interaction Based on Dynamic Bayesian Networks Probabilistic Intention Recognition," *Journal of Intelligent and Robotic Systems*, vol. 45, pp. 31-52, 2006.
85. P. A. Jarvis, T. F. Lunt, and K. L. Myers, "Identifying Terrorist Activity with AI Plan Recognition Technology," in the Sixteenth National Conference on Innovative Applications of Artificial Intelligence 2004, pp. 858--863.
86. F. Mulder and F. Voorbraak, "A formal description of tactical plan recognition," *Information Fusion*, vol. 4, pp. 47-61, 2003.
87. F. Toni, P. Torroni, R. Demolombe, and A. Fernandez, "Intention Recognition in the Situation Calculus and Probability Theory Frameworks," in *Computational Logic in Multi-Agent Systems*. vol. 3900: Springer Berlin Heidelberg, 2006, pp. 358-372.
88. P. Quaresma and J. G. Lopes, "Unified logic programming approach to the abduction of plans and intentions in information-seeking dialogues," *The Journal of Logic Programming*, vol. 24, pp. 103-119, 1995.
89. L. M. Pereira and A. Han, "Elder care via intention recognition and evolution prospection," in *proceedings of the 18th international conference on Applications of declarative programming and knowledge management*, Vora, Portugal, 2009.
90. L. M. Pereira and A. Saptawijaya, "Modelling morality with prospective logic," *International Journal of Reasoning-based Intelligent Systems*, vol. 1, pp. 209-221, 2009.
91. A. F. Dragoni, P. Giorgini, and L. Serafini, "Mental States Recognition from Communication," *Journal of Logic Computation*, vol. 12, pp. 119-136, 2002.
92. E. Charniak and R. P. Goldman, "A Bayesian model of plan recognition," *Artif. Intell.*, vol. 64, pp. 53-79, 1993.
93. K. A. Tahboub, "Intelligent human-machine interaction based on dynamic Bayesian networks probabilistic intention recognition," *Journal*

- of Intelligent and Robotic Systems: Theory and Applications, vol. 45, pp. 31-52, 2006.
94. K. A. Tahboub, "Intention recognition of a human commanding a mobile robot," in IEEE Conference on Cybernetics and Intelligent Systems, 2004, pp. 896-901.
 95. L. He, C.-f. Zong, and C. Wang, "Driving intention recognition and behaviour prediction based on a double-layer hidden Markov model," *Journal of Zhejiang University SCIENCE C*, vol. 13, pp. 208-217, 2012.
 96. L. J. Haijing Hou, Qingning Niu, Yuqin Sun, Meng Lu "Driver Intention Recognition Method Using Continuous Hidden Markov Model " *International Journal of Computational Intelligence Systems* vol. 4, pp. 386-393, May 2011 2011.
 97. D. Aarno and D. Kragic, "Layered HMM for Motion Intention Recognition," in IEEE/RSJ International Conference on Intelligent Robots and Systems, 2006, pp. 5130-5135.
 98. Q. Zhang, D. Man, and W. Yang, "Using HMM for Intent Recognition in Cyber Security Situation Awareness," in *Proceedings of the 2009 Second International Symposium on Knowledge Acquisition and Modeling - Volume 02*, Wuhan, China, 2009, pp. 166-169.
 99. N. Oliver, A. Garg, and E. Horvitz, "Layered representations for learning and inferring office activity from multiple sensory channels," *Computer Vision and Image Understanding*, vol. 96, pp. 163-180, Nov 2004.
 100. J. Yang and Y. Xu, "Hidden markov model for gesture recognition," *DTIC Document*, May1994.
 101. T.-H. Nguyen, D. Hsu, W.-S. Lee, T.-Y. Leong, L. Kaelbling, T. Lozano-Perez, and A. Grant, "CAPIR: Collaborative Action Planning with Intention Recognition," in *Proceedings of the Seventh Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE 2011)*, Palo Alto, California, 2011, pp. 61-66.
 102. D. Holmes, L. Jain, and D. Niedermayer, "An Introduction to Bayesian Networks and Their Contemporary Applications," in *Innovations in Bayesian Networks*. vol. 156: Springer Berlin Heidelberg, 2008, pp. 117-130.
 103. C. Heinze, "Modeling Intention Recognition for Intelligent Agent Systems," in *Department of Defence*. vol. Doctoral Thesis: the University of Melbourne, 2003, p. 236.
 104. M. C. Vassiliou, L. S. Feldman, C. G. Andrew, S. Bergman, K. Leffondre, D. Stanbridge, and G. M. Fried, "A global assessment tool for

evaluation of intraoperative laparoscopic skills," *Am J Surg*, vol. 190, pp. 107-113, Jul 2005.

105. F. Chuan, L. Han, J. Rozenblit, J. Peng, a. Hamilton, and M. Salkini, "Surgical training and performance assessment using a motion tracking system," in the 2nd European Modeling and Simulation Symposium, Piera: LogiSim, Barcelona, Spain, 2006, pp. 647-652.
106. L. Zhuohua, M. Uemura, M. Zecca, S. Sessa, H. Ishii, M. Tomikawa, M. Hashizume, and A. Takanishi, "Objective Skill Evaluation for Laparoscopic Training Based on Motion Analysis," *IEEE Transactions on Biomedical Engineering*, vol. 60, pp. 977-985, 2013.
107. Z. Qiang and L. Baoxin, "Relative Hidden Markov Models for Evaluating Motion Skill," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 548-555.
108. G. Megali, S. Sinigaglia, O. Tonet, and P. Dario, "Modelling and evaluation of surgical performance using hidden Markov models," *IEEE Trans Biomed Eng*, vol. 53, pp. 1911-9, Oct 2006.
109. J. Rosen, B. Hannaford, C. G. Richards, and M. N. Sinanan, "Markov modeling of minimally invasive surgery based on tool/tissue interaction and force/torque signatures for evaluating surgical skills," *IEEE Transactions on Biomedical Engineering*, vol. 48, pp. 579-591, 2001.
110. J. Rosen, M. MacFarlane, C. Richards, B. Hannaford, and M. Sinanan, "Surgeon-tool force/torque signatures--evaluation of surgical skills in minimally invasive surgery," *Stud Health Technol Inform*, vol. 62, pp. 290-6, 1999.
111. J. Rosen, M. Solazzo, B. Hannaford, and M. Sinanan, "Objective laparoscopic skills assessments of surgical residents using Hidden Markov Models based on haptic information and tool/tissue interactions," *Stud Health Technol Inform*, vol. 81, pp. 417-23, 2001.
112. F. Chuan, J. W. Rozenblit, and A. Hamilton, "Fuzzy Logic-Based Performance Assessment in the Virtual, Assistive Surgical Trainer (VAST)," in *Engineering of Computer Based Systems*, 2008. ECBS 2008. 15th Annual IEEE International Conference and Workshop on the, 2008, pp. 203-209.
113. I. Laptev and T. Lindeberg, "Space-time interest points," in *Proceedings Ninth IEEE International Conference on Computer Vision*, 2003, pp. 432-439 vol.1.
114. T. Yang, J. Liu, W. Huang, Y. Su, L. Yang, C. Chui, M. Ang, Jr., and S. Y. Chang, "Mechanism of a Learning Robot Manipulator for Laparoscopic Surgical Training," in *Intelligent Autonomous Systems 12*, vol. 194: Springer Berlin Heidelberg, 2012, pp. 17-26.

115. J. Zhou, W. Huang, J. Zhang, T. Yang, J. Liu, C. K. Chui, and K. Y. S. Chang, "Segmentation of gallbladder from CT images for a surgical training system," in 2010 3rd International Conference on Biomedical Engineering and Informatics (BMEI), 2010, pp. 536-540.
116. C. L. Gim Han Law Melvin Eng, Su Yi, Weimin Huang, Jiayin Zhou, Jiang Liu, Jing Zhang, Tao Yang, Chee Kong Chui, Stephen Chang "Rapid Generation of Patient-Specific Anatomical Models for Usage in Virtual Environment," Computer Aided Design and Application, vol. 8, pp. 927-938, 2011.
117. R. Vijaykumar, M. Tsai, and K. Waldron, "Geometric optimization of manipulator structures for working volume and dexterity," in Proceedings IEEE International Conference on Robotics and Automation, 1985, pp. 228-236.
118. O. Piccin, B. Bayle, B. Maurin, and M. de Mathelin, "Kinematic modeling of a 5-DOF parallel mechanism for semi-spherical workspace," Mechanism and Machine Theory, vol. 44, pp. 1485-1496, 2009.
119. T. Yang, L. Xiong, J. Zhang, L. Yang, W. Huang, J. Zhou, J. Liu, Y. Su, C. K. Chui, C. L. Teo, and S. Chang, "Modeling cutting force of laparoscopic scissors," in 3rd International Conference on Biomedical Engineering and Informatics (BMEI), 2010, pp. 1764-1768.
120. H. Olsson, K. J. Astrom, C. Canudas de Wit, M. Gafvert, and P. Lischinsky, "Friction Models and Friction Compensation," European Journal of Control, vol. 4, pp. 176-195, 1998.
121. B. Armstrong-HÃ©louvy, P. Dupont, and C. C. De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," Automatica, vol. 30, pp. 1083-1138, 1994.
122. M. P. Wand and M. C. Jones, Kernel Smoothing. Londong: Chapman & Hall, 1995.
123. A. R. Mugdadi and I. A. Ahmad, "A bandwidth selection for kernel density estimation of functions of random variables," Computational Statistics & Data Analysis, vol. 47, pp. 49-62, 2004.
124. S. J. Sheather and M. C. Jones, "A Reliable Data-Based Bandwidth Selection Method for Kernel Density Estimation," Journal of the Royal Statistical Society, vol. 53, pp. 683-690, 1991.
125. I. Horová, J. Koláček, J. Zelinka, and K. Vopatová, "Bandwidth choice for kernel density estimates," in 6th Conference of the Asian Regional Section of the IASC, Yokohama Japan, 2008.

126. D. Comaniciu, V. Ramesh, and P. Meer, "The variable bandwidth mean shift and data-driven scale selection," in 8th IEEE International Conference on Computer Vision, 2001, pp. 438-445 vol.1.
127. I. Horová, J. Kolář, and K. Vopatová, "Full bandwidth matrix selectors for gradient kernel density estimate," Computational Statistics & Data Analysis, vol. 57, pp. 364-376, 2012.
128. R. M. Kho, "Comparison of robotic-assisted laparoscopy versus conventional laparoscopy on skill acquisition and performance," Clin Obstet Gynecol, vol. 54, pp. 376-81, Sep 2011.
129. G. Megali, S. Sinigaglia, O. Tonet, and P. Dario, "Modelling and Evaluation of Surgical Performance Using Hidden Markov Models," IEEE Transactions on Biomedical Engineering, vol. 53, pp. 1911-1919, 2006.
130. <http://programming-by-demonstration.org>

AUTHOR'S PUBLICATION

Patents:

A Robotic Device for Use in Image-Guided Robot Assisted Surgical Training, United States Patent 8,764,448, 1st July 2014.

Inventors: T. Yang; L. Yang; J. Liu; C. K. Chui; W. Huang; J. Zhang; J. Zhou; B. H. Lee; N. M. Tan; W. K. Wong; F. Yin; S. K. Chang; Y. Su.

Journal Publications

1. T. Yang, C. K. Chui, J. Liu, W. Huang, Y. Su, and S. Chang, "Robotic learning of motion using demonstrations and statistical models for surgical simulation," Int J Comput Assist Radiol Surg, vol. 9, pp. 813-23, Sep.2013
2. J. Zhang, J. Zhou, W. Huang, J. Qin, T. Yang, J. Liu, Y. Su, C. K. Chui, and S. Chang, "GPU-friendly Gallbladder Modelling in Laparoscopic Cholecystectomy Surgical Training System," Computers & Electrical Engineering, vol. 39, pp. 122-129, Jan 2013.
3. T. Yang, C. K. Chui, J. Liu and W. Huang, Motion Intention Recognition in Laparoscopic Surgical Training (to be submitted to IEEE Transactions on Human-Machine System).

Conference Publications

1. Y. Duan, W. Huang, H. Chang, W. Chen, K. Toe, J. Zhou, T. Yang, J. Liu, S. Teo, C. Lim, Y. Su, C. Chui, and S. Chang, "Modelling and Simulation of Soft Tissue Deformation," in Abdominal Imaging. Computation and Clinical Applications. vol. 8198: Springer Berlin Heidelberg, pp. 221-230, 2013.
2. Y. Duan, W. Huang, H. Chang, K. Toe, T. Yang, J. Zhou, J. Liu, S. K. Teo, C. W. Lim, Y. Su, C. K. Chui, and S. Chang, "Synchronous simulation for deformation of liver and gallbladder with stretch and compression compensation," Conf Proc IEEE Eng Med Biol Soc, vol. 2013, pp. 4941-4.
3. T. Yang, K. Toe, C. Chng, W. Huang, C. K. Chui, J. Liu, and S. Chang, "Evaluation of Image Guided Robot Assisted Surgical Training for Patient Specific Laparoscopic Surgery," in Bio-Imaging and Visualization for Patient-Customized Simulations. vol. 13: Springer International Publishing, pp. 77-88, 2013

4. T. Yang, J. Liu, W. Huang, Y. Su, L. Yang, C.K. Chui, M. Ang, Jr., and S. Y. Chang, "Mechanism of a Learning Robot Manipulator for Laparoscopic Surgical Training," in *Intelligent Autonomous Systems 12*, vol. 194: Springer Berlin Heidelberg, pp. 17-26, 2012
5. C.K. Chui, C.B. Chng, Y. Tao, R. Wen, W. Huang, J. Liu, Y. Su, and S. Chang, "Learning laparoscopic surgery by imitation using robot trainer," in *2011 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2011, pp. 2981-2986.
6. S. Lee, L. Yang, T. Yang, C. K. Chui, J. Liu, W. Huang, Y. Su, and S. Chang, "Designing an active motor skill learning platform with a robot-assisted laparoscopic trainer," *Conf Proc IEEE Eng Med Biol Soc*, vol. 2011, pp. 4534-7, 2011.
7. J. Qin, J. Zhang, C.K. Chui, W.M. Huang, T. Yang, W.M. Pang, V. Sudhakar and S. Chang, "A simulation framework for estimating wall stress distribution of abdominal aortic aneurysm," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC*, 2011, pp. 900-903.